



PRACTICAS DE SERVICIOS DE RED E INTERNET

En este libro se abarcará la asignatura de servicios de red e internet del grado superior de informática.

Escrito por: Nicolás Madrid Gallego

Nicolás Madrid Gallego
IES GREGORIO PRIETO
HTTP



Contenido

Actividad 1 : W3C y estándares Web. Servidores Web. Clientes Web (navegadores).	5
Consulta la web www.w3.org del W3C y observa los estándares que desarrolla. Pincha en el enlace W3C A to Z para tener una visión general de todos los estándares de la Web. Después accede a la web en español www.w3c.es . Consulta el hiperenlace Documentos y guías que muestra el material en español basado en las especificaciones técnicas del W3C, las Guías breves y el Glosario de la A a la Z. Elabora un informe con dicha documentación de tecnologías W3C y estándares Web.	5
b) Existe cierta confusión en la comunidad web sobre la relación de los términos URI, URL y URN (<i>Uniform Resource Identifier</i>). Amplia dicha información, profundizando sobre dichos formatos consultando la web http://www.w3.org/Addressing/ de la W3C o investiga en Internet hasta tener claro dichos conceptos. Elabora un breve documento sobre la relación entre dichos términos.	18
c) Visita la web http://netcraft.com/ que ofrece servicios de análisis y comparativas de la Web. En su enlace “What’s that site running” se permite consultar información sobre el servidor web en el que se ejecuta o aloja un sitio web determinado. Indica en un informe los servidores web que dan soporte a tus sitios web preferidos. Por ejemplo: www.iesgregorioprieto.org , www.jccm.es , ...etc. Incluye asimismo el gráfico correspondiente al ranking de Servidores Web utilizados en Internet.	19
d) Inicia sesión con el navegador Firefox en una distribución GNU/Linux. Accede en la barra de menú a herramientas, complementos. Observa las extensiones, temas, idiomas y <i>plugins</i> instalados. Haz clic en obtener complementos y verás una lista de complementos disponibles para descargar e instalar. Instala al menos uno. Documenta dicho proceso. Si no tienes instalado dicho navegador accede a su web (http://www.mozilla-europe.org/es/) e instala el mismo.	21
e) Realiza y documenta la instalación de <i>plugins</i> para los navegadores Internet Explorer (instalado por defecto en Windows) y Google Chrome (http://www.google.es/chrome). Si necesitas instalar dichos navegadores visita sus sitios web de descarga.	26
f) Instala los navegadores Safari (http://www.apple.com/es/safari/) Opera (http://www.opera.com), Amaya (http://www.w3.org/Amaya/) y el navegador de texto Lynx (http://lynx.isc.org/). Documenta con ayuda de la Wikipedia la historia de dichos navegadores web, sus características y compara los mismos. Comienza consultando el sitio web: http://es.wikipedia.org/wiki/Navegador_web y http://es.wikipedia.org/wiki/Anexo:Tabla_comparativa_de_navegadores_web . Indica cuál es para ti el mejor de todos los navegadores web existentes en el mercado informático . Razona tu respuesta.	30
g) Comprueba e investiga en al menos dos navegadores web de los citados anteriormente la configuración de las cookies. Documenta dicho proceso.	35
Actividad 3 : Instalación y configuración de un servidor HTTP (Apache2) en Ubuntu Server (configuraciones por defecto).	42
Actividad 5: Administración servidor Web HTTP (Apache2) en Ubuntu Server : Directorio por Defecto “Directory Index” -- Opciones sobre Directorios --Logs --Códigos de error.	47

Actividad 6: Administrador servidor Web HTTP (Apache2) en Ubuntu Server: -- Directorios Virtuales (<i>Alias</i> y <i>enlaces</i> simbólicos).....	54
Actividad 7: Administración servidor Web HTTP (Apache2) en Ubuntu Server: --Módulos -- Directorios personales de usuarios	57
Actividad 8: Administración servidor Web HTTP (Apache2) en Ubuntu Server: --Control de acceso por IP -- Autenticación HTTP Basic.	62
Actividad 9: Administración servidor Web HTTP (Apache2) en Ubuntu Server: --Fichero .htaccess.....	66
Actividad 10: Administración servidor Web HTTP (Apache2) en Ubuntu Server: -- Monitorización y logs -- Pruebas de rendimiento.	69
Actividad 11: Administración servidor Web HTTPS (Apache2) en Ubuntu Server: -- Escenario: Alojamiento virtual basado en IPs.	78
Actividad 12: Administración servidor Web HTTPS (Apache2) en Ubuntu Server --Escenario: Alojamiento virtual basado en nombres.....	81
Actividad 13: Administración servidor Web HTTPS (Apache2) en Ubuntu Server --Escenario: Alojamiento virtual basado en puertos.....	85
Actividad 14: Instalación y configuración de un servidor web HTTP Apache2 mediante Webmin.	88
Actividad 15: Administración servidor web HTTPS (Apache2) en Ubuntu Server: -- mod_ssl, default_ssl – certificados digitales.....	97

Actividad 1 : W3C y estándares Web. Servidores Web. Clientes Web (navegadores).

Consulta la web www.w3.org del W3C y observa los estándares que desarrolla. Pincha en el enlace [W3C A to Z](#) para tener una visión general de todos los estándares de la Web. Después accede a la web en español www.w3c.es . Consulta el hiperenlace [Documentos y guías que muestra el material en español basado en las especificaciones técnicas del W3C, las Guías breves y el Glosario de la A a la Z. Elabora un informe con dicha documentación de tecnologías W3C y estándares Web.](#)

Accesibilidad Web

Hablar de *Accesibilidad Web* es hablar del acceso de todos a la Web, independientemente del tipo de hardware, software, infraestructura de red, idioma, cultura, localización geográfica y capacidades de los usuarios.

Con esta idea de accesibilidad nace la Iniciativa de Accesibilidad Web, conocida como WAI (Web Accessibility Initiative), una actividad desarrollada por el W3C, cuyo objetivo principal es facilitar el acceso a la Web a todas aquellas personas con discapacidad, desarrollando pautas de accesibilidad, mejorando las herramientas para la evaluación y reparación de accesibilidad Web, llevando a cabo una labor educativa y de concienciación en relación a la importancia del diseño accesible de páginas Web y abriendo nuevos campos en accesibilidad a través de la investigación en este área.

Amaya

Amaya es una herramienta que permite al mismo tiempo navegar, editar y publicar información en la Web. La integración de estas tres funciones hace más fácil para los usuarios participar en la aportación de contenido para la Web. Su objetivo principal es proporcionar una infraestructura fácilmente extensible que sirva como marco común de integración del mayor número posible de tecnologías del W3C para así poder experimentar y realizar demostraciones combinando estas tecnologías.

Annotea

Annotea es un sistema que permite la creación y publicación en la Web de anotaciones compartidas de recursos. Una *anotación* puede ser una nota, una explicación u otro tipo de comentario externo que puede agregarse a cualquier documento Web o a una parte seleccionada, sin necesidad de modificar el documento. Su principal aportación es la mejora de la colaboración de grupos de trabajo a través de esas anotaciones Web compartidas.

APPEL

El *Lenguaje de Intercambio de Preferencias P3P (P3P Preference Exchange Language)*, es un lenguaje para describir colecciones de preferencias relacionadas con las directivas P3P entre distintos agentes. Usando este lenguaje, un usuario puede expresar sus preferencias a través de un conjunto de reglas, las cuales pueden ser utilizadas por el agente de usuario para tomar decisiones automáticas o semiautomáticas de acuerdo a la aceptación de las políticas de privacidad de los sitios Web, en función de lo que el usuario prefiera.

Buenas Prácticas en Web Móvil

Las *Buenas Prácticas para Web Móvil* son un conjunto de pautas para la creación y publicación de contenido Web para dispositivos móviles. El objetivo de estas directrices es ayudar en el desarrollo de un sitio para favorecer el acceso correcto de los usuarios que usan dispositivos como PDAs, teléfonos móviles, u otros dispositivos con características mucho más limitadas que un equipo de escritorio.

Coreografía de Servicios Web

Una *Coreografía de Servicios Web* es una representación conjunta de la funcionalidad de un conjunto de Servicios Web, que interactúan entre sí para componer un servicio homogéneo final. El *Lenguaje de Descripción de Coreografías de Servicios Web (Web Services Choreography Description Language o WS-CDL)* permite definir las relaciones entre los servicios y representar la funcionalidad final tras la composición de éstos.

CC/PP

Capacidad de Composición/Perfiles de Preferencia (Composite Capability/Preference Profiles). Es un sistema basado en RDF, que permite estandarizar la forma de definir y transmitir información sobre las características de los dispositivos y las preferencias del usuario. El objetivo es facilitar la adaptación del contenido al que el usuario desea acceder tomando como referente el tipo de dispositivo utilizado y las características del usuario.

CCXML

Lenguaje de Etiquetado Extensible de Control Telefónico (Call Control eXensible Markup Language). Es un lenguaje diseñado para proporcionar soporte para el control de llamadas telefónicas que se puede utilizar tanto con VoiceXML, como con otros sistemas de diálogo. Se podría decir que CCXML es un complemento de VoiceXML, que proporciona funciones de telefonía avanzadas que van más allá de la capacidad que tiene VoiceXML.

CSS

Hojas de Estilo en Cascada (Cascading Style Sheets). Es un mecanismo para dar estilo a documentos HTML y XML, que consiste en reglas simples a través de las cuales se establece cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores control total sobre el estilo y formato de sus documentos separando contenido y presentación.

DIAL

El *Lenguaje de creación de contenido Independiente del Dispositivo (Device Independent Authoring Language)* permite estructurar un documento para filtrar y representar el contenido de una página Web disponible en distintos contextos. Esto asegura que el contenido se adapta a las posibilidades del usuario y su dispositivo, así como a las circunstancias en las que se encuentre.

Direccionamiento de Servicios Web

Direccionamiento (Addressing) de Servicios Web son unos mecanismos que permiten dirigir los Servicios Web y sus mensajes por las redes. Mediante esta tecnología se puede realizar representaciones que hacen referencia a servicios y facilitan el direccionamiento entre los puntos finales de los mensajes de los servicios Web, que pueden transmitirse por la red a través de distintos nodos de procesamiento (cortafuegos, gestores finales, o pasarelas) siempre usando una forma de transporte neutra.

Directivas de Servicios Web

El lenguaje de *Directivas de Servicios Web (Web Services Policy)* permite describir y combinar las características y requisitos de un servicio Web. Esto permite la escalabilidad de los servicios, así como una definición más completa y precisa de los mismos.

Documentos Compuestos

Un *Documento Compuesto (Compound Document)* es un documento que combina múltiples formatos, como puede ser XHTML, SVG, SMIL y XForms. Estas combinaciones, con especificaciones concretas, hacen posible una Web escalable e interoperable.

DOM

Modelo de Objetos del Documento (Document Object Model). Es una plataforma que proporciona un conjunto estándar de objetos a través de la cual se pueden crear documentos HTML y XML, navegar por su

estructura y, modificar, añadir y borrar tanto elementos como contenidos. Al no apoyarse en un lenguaje de programación en particular, DOM facilita el diseño de páginas Web activas, proporcionando una interfaz estándar para que otro software manipule los documentos.

EXI

El *Intercambio de XML Eficiente (Efficient XML Interchange)* es un formato para la representación muy compacta del conjunto de información de XML, que pretende optimizar tanto el rendimiento, como la utilización de los recursos de forma simultánea.

EMMA

Lenguaje de Anotación Multimodal Extensible (Extensible MultiModal Annotation language). Es un lenguaje estándar utilizado por los componentes que intervienen en un sistema multimodal para comunicarse. Su objetivo es integrar las entradas de datos proporcionadas por los usuarios desde diferentes recursos y darles forma para que así esos datos sean procesados como una representación única. Dentro de una infraestructura de Interacción Multimodal, EMMA actúa como un mecanismo de intercambio entre los dispositivos a través de los cuales el usuario introduce datos.

GRDDL

GRDDL es un mecanismo para *Recuperar Descripciones de Recursos de Dialectos de Lenguajes (Gleaning Resource Descriptions from Dialects of Languages)*. GRDDL incluye etiquetado para indicar que un documento XML incluye información que se puede obtener y especifica la notación especial a incluir para enlazar un mecanismo que permita extraer las descripciones de los recursos del documento. Una utilidad de esta tecnología es la recolección de descripciones en formato RDF, desde el contenido de un fichero XHTML.

HTML

Lenguaje de Etiquetado de Hipertexto (HyperText Markup Language). Es un lenguaje comúnmente utilizado para la publicación de hipertexto en la Web y desarrollado con la idea de que cualquier persona o tipo de dispositivo pueda acceder a la información en la Web. HTML utiliza etiquetas que marcan elementos y estructuran el texto de un documento.

HTTP

Protocolo de Transferencia de Hipertexto (Hypertext Transfer Protocol). Es un protocolo utilizado para la transferencia de datos a través de Internet, y que está basado en operaciones sencillas de solicitud y respuesta.

Independencia de Dispositivo

La *Independencia de Dispositivo* facilita el acceso a la información independientemente del dispositivo utilizado para acceder a la información. Ésta va a estar siempre disponible y accesible para el usuario, es decir, se trata de hacer la Web universal y accesible para cualquier persona, en cualquier sitio, en cualquier momento y usando cualquier dispositivo, evitando la fragmentación de la Web en espacios accesibles sólo por dispositivos concretos. El objetivo principal es mejorar la experiencia del usuario y al mismo tiempo reducir costes al desarrollar estándares que permitan acceder a esa información desde cualquier dispositivo.

InkML

Lenguaje de Etiquetado de Tinta (Ink Markup Language). Es un lenguaje XML diseñado para facilitar el intercambio de escritura electrónica entre distintas aplicaciones. InkML captura movimientos, ángulo y presión de forma electrónica para su posterior almacenaje y reconocimiento. No sólo reconoce escritura (texto plano, matemáticas, química, etc.), sino que verifica firmas e interpreta gestos.

Interacción multimodal

La *Interacción Multimodal* o *Multimodalidad* consiste en un proceso en el cual diversos dispositivos y personas son capaces de llevar a cabo una interacción conjunta (auditiva, visual, táctil y gestual) desde cualquier sitio, en cualquier momento, utilizando cualquier dispositivo y de forma accesible, mejorando así la interacción entre personas, y entre dispositivos y personas.

Internacionalización

La *Internacionalización* es un concepto que implica un diseño Web sin barreras para los diferentes idiomas, sistemas de escritura, códigos de caracteres y otras convenciones locales que existen. El carácter global de la Web requiere un sistema en el que exista facilidad a la hora de crear y procesar información para una audiencia variada, permitiendo así publicar material e intercambiar datos en cualquier idioma. La Internacionalización también es conocida como I18N, que es la abreviatura de Internacionalización ya que entre la "i" y la "n" existen 18 letras.

JigSaw

JigSaw es una plataforma de servidor Web basada en Java cuyo principal objetivo es hacer demostraciones de nuevos protocolos y al mismo tiempo ofrecer una base de experimentación en el área de software para servidor.

Libwww

Libwww es una interfaz de programación de aplicaciones (API) Web de cliente, de uso común y alta modularidad, cuyo principal propósito es el de servir como herramienta de pruebas para experimentos con protocolos.

MathML

Lenguaje de Etiquetado Matemático (Mathematical Markup Language). Es un lenguaje cuyo objetivo principal es permitir que las expresiones matemáticas puedan ser representadas y procesadas en la Web al igual que el HTML lo hace con el texto simple. Consiste en etiquetas XML que pueden utilizarse para definir ecuaciones en términos de su presentación y su semántica.

MobileOK

mobileOK Básico es un esquema para evaluar si el contenido de una Web puede ser enviado de forma que cumple con las Buenas Prácticas para Web Móvil. Para la evaluación se utiliza un agente de usuario móvil hipotético. El objetivo de mobileOK es ayudar a desarrollar más rápidamente contenido Web móvil.

OWL

Lenguaje de Ontologías Web (Web Ontology Language). Una ontología define los términos a utilizar para describir y representar un área de conocimiento. Las ontologías son utilizadas por las personas, las bases de datos, y las aplicaciones que necesitan compartir un dominio de información (un dominio es simplemente un área de temática específica o un área de conocimiento, tales como medicina, bienes inmuebles, gestión financiera, etc.). Las ontologías incluyen definiciones de conceptos básicos del dominio, y las relaciones entre ellos, que son útiles para los ordenadores. OWL permite definir una ontología en términos de XML.

PICS

Plataforma para la Selección de Contenido en Internet (Platform for Internet Content Selection). Es un conjunto de especificaciones técnicas que ofrecen al usuario la posibilidad de realizar un acceso selectivo a la información que se puede obtener en la Web; esto se consigue a través de la asociación de etiquetas (metadatos) a contenido de Internet, funcionando como un filtro.

PNG

Gráficos de Red Portátiles (Portable Network Graphics). Es uno de los formatos gráficos más utilizados en la Web ya que permite un

almacenamiento sin pérdida, portátil y con buena compresión de gráficos. Soporta imágenes en color indexado, en escala de grises y color verdadero, además de varios niveles de transparencia.

Política de Patentes

Se trata de una serie de normas a través de las cuales el W3C establece la forma de gestionar las patentes en el proceso de creación y desarrollo de estándares Web con el objetivo de hacer posible la difusión, adopción y continuo desarrollo de estándares Web libres de derechos de autor.

P3P

Plataforma de Preferencias de Privacidad (Platform for Privacy Preferences). Es un lenguaje estándar que ofrece a los usuarios una forma sencilla y automatizada de controlar en mayor medida el uso que se hace de su información personal en los sitios Web que visitan. Las prácticas de privacidad de los sitios Web a través de P3P adquieren un formato estándar que los navegadores con P3P podrán interpretar automáticamente comparando esas prácticas de privacidad con las preferencias de privacidad del usuario, con lo que el usuario podrá decidir si quiere o no, y bajo qué circunstancias, revelar información personal.

RDF

Infraestructura para la Descripción de Recursos (Resource Description Framework). Es un formato universal para representar datos en la Web, que va a permitir intercambiar información a través de diferentes aplicaciones sin que esos datos pierdan significado, lo que facilita la reutilización de los recursos en la Web.

RDFa

RDFa permite incluir datos estructurados (eventos de calendario, datos de contacto, información sobre las licencias o los derechos de autor, etc.) dentro de las páginas Web, de forma que otras aplicaciones puedan transferir y recopilar la información de forma eficiente. Por ejemplo, gracias a RDFa, una aplicación de calendario podría recopilar los datos que aparecen en un sitio web, cuando se navega por él.

RIF

El *Formato de Intercambio de Reglas* (Rule Interchange Format) es un mecanismo que permite la comunicación entre antiguos sistemas basados en reglas de acuerdo con procesos y metodologías basadas en estándares. La finalidad de este formato es asegurar que los sistemas complejos no sólo funcionen en sus entornos locales, sino que puedan hacerse extensibles con la evolución de las tecnologías de reglas y sean interoperables con otras tecnologías.

SISR

La *Interpretación Semántica para el Reconocimiento del Habla* (*Semantic Interpretation for Speech Recognition*) es una parte de los navegadores por voz que permiten definir el proceso para la interpretación del reconocimiento del habla, así como la sintaxis y semántica del contenido de las etiquetas en la Especificación de Gramática de Reconocimiento del Habla (SRGS). Los resultados de la interpretación semántica, describen el significado de las palabras en lenguaje natural.

SML

Lenguaje de Modelado de Servicio (*Service Modeling Language*) se utiliza para modelar servicios y sistemas complejos incluyendo su estructura, restricciones, directivas y buenas prácticas. Se basa en el establecimiento de esquemas de documentos y de reglas para especificar y limitar la estructura y el contenido de los documentos.

SMIL

Lenguaje de Integración de Multimedia Sincronizada (*Synchronized Multimedia Integration Language*). Es un lenguaje de etiquetado utilizado para crear presentaciones multimedia dinámicas a través de la sincronización en tiempo y espacio de diversos elementos como audio, vídeo, texto y gráficos. Permite a los desarrolladores especificar y controlar el momento preciso en el que se ha de integrar audio y vídeo con imágenes, texto y otros tipos de medios, permitiendo, al mismo tiempo, la posibilidad de interacción con el usuario.

Servicios Web

Los *Servicios Web* son un conjunto de aplicaciones o de tecnologías con capacidad para interoperar en la Web y que intercambian datos entre sí con el objetivo de ofrecer servicios. Los proveedores ofrecen sus servicios como procedimientos remotos y los usuarios solicitan un servicio llamando a estos procedimientos a través de la Web. Estos servicios proporcionan mecanismos de comunicación estándares entre diferentes aplicaciones, que interactúan entre sí para presentar información dinámica al usuario.

SOAP

Protocolo Simple de Acceso a Objetos (*Simple Object Access Protocol*). Es un protocolo para intercambiar mensajes, basado en XML, y de extendido uso en Servicios Web.

SPARQL

El *Protocolo Simple y Lenguaje de Consulta de RDF (Simple Protocol And RDF Query Language)* es una tecnología que permite hacer consultas sobre información expresada en RDF, usando distintas fuentes de datos y permite obtener también los resultados en formato RDF para poder utilizarlos en diversas aplicaciones.

SSML

Lenguaje de Etiquetado de Síntesis del Habla (Speech Synthesis Markup Language). Es un lenguaje de etiquetado, basado en XML, que forma parte de un conjunto de especificaciones para navegadores por voz que ayuda a generar habla sintética en la Web y en otras aplicaciones. Proporciona un camino estándar para controlar aspectos del habla como son pronunciación, volumen, tono y velocidad.

SRGS

Especificación de Gramática de Reconocimiento del Habla (Speech Recognition Grammar Specification). Es una especificación que trata de definir la sintaxis necesaria para representar gramáticas utilizadas en el reconocimiento del habla. De esta forma, los desarrolladores pueden especificar las palabras o grupos de palabras que un reconocedor del habla debe captar. Un reconocedor es capaz de establecer una relación entre el audio proporcionado por el usuario y una gramática, generando así una transcripción.

SVG

Gráficos Vectoriales Escalables (Scalable Vector Graphics). Es un lenguaje de descripción de gráficos de dos dimensiones y de aplicaciones gráficas en XML. SVG se utiliza en diferentes áreas incluyendo gráficos Web, animación, interfaces de usuario, intercambio de gráficos, aplicaciones móviles y diseño de alta calidad.

Texto Temporizado

La especificación de *Texto Temporizado (Timed-Text)* en la Web cubre los aspectos necesarios para incluir textos sincronizados para distintas aplicaciones en la Web, como subtítulos de películas, transcripciones para equipos sin audio, karaokes, titulares de noticias, u otras.

URI

Identificador de Recurso Uniforme (Uniform Resource Identifiers). Los URI son cadenas que funcionan como identificadores globales que hacen referencia a recursos en la Web tales como documentos, imágenes, archivos descargables, servicios, buzones de correo electrónico y otros.

VoiceXML

Lenguaje de Etiquetado Extensible de Voz (Voice Extensible Markup Language). Es un lenguaje para crear diálogos entre el usuario y la aplicación. La aplicación se comunica con el usuario enviando información hablada, y el usuario interactúa con la misma ejecutando acciones a través de la voz.

WebCGM

WebCGM (Web Computer Graphics Metafile). WebCGM es un formato de archivo binario que define un subconjunto del estándar CGM para dar soporte a las funcionalidades de: hipervinculación y navegación de documentos, estructura y organización en capas, e investigación y peticiones en contenido de imagen WebCGM. Una de las principales ventajas de WebCGM es que es compatible con Unicode y todos los formatos Web.

Web Móvil

La *Web Móvil* es una Web accesible desde cualquier lugar a través de dispositivos móviles. El W3C con el objetivo de hacer el acceso a la Web desde estos dispositivos algo sencillo y cómodo ha puesto en marcha la *Iniciativa de Web Móvil* que busca resolver los problemas de interoperabilidad y usabilidad que actualmente dificultan el acceso a la Web desde dispositivos móviles.

Web Semántica

La *Web Semántica* es una Web extendida y dotada de mayor significado, apoyada en lenguajes universales, que van a permitir que los usuarios puedan encontrar respuestas a sus preguntas de forma más rápida y sencilla gracias a una información mejor definida. Con esta Web, los usuarios podrán delegar tareas en el software que será capaz de procesar el contenido de la información, razonar con éste, combinarlo y realizar deducciones lógicas para resolver automáticamente problemas cotidianos.

Widget

Un Widget es un tipo de aplicación Web en la parte cliente que permite representar y/o actualizar datos locales o remotos. Uno de los beneficios de los Widgets es que están empaquetados para ofrecer su descarga y su instalación de forma sencilla en los dispositivos desde los que se accede. Ejemplos de estos Widgets son: relojes, indicadores bursátiles, previsiones climatológicas, noticias de última hora, etc.

WSDL

El *Lenguaje de Descripción de Servicios Web (Web Services Description Language)* permite definir lo que hace un Servicio Web según la funcionalidad que ofrece. Mediante este lenguaje se representa la interfaz de uso del servicio, lo que tendrán que tener en cuenta otros servicios a la hora de acceder a su funcionalidad.

XBL

El *Lenguaje de Vinculación de XML (XML Binding Language)* permite asociar elementos de un documento con scripts, manejadores de eventos, hojas de estilo CSS y otros modelos de contenido complejos que pueden estar almacenados en otro documento. Esto puede ser utilizado, por ejemplo, para reorganizar el contenido de un documento, modificando el etiquetado XHTML para aplicar hojas de estilo CSS complejas sin ensuciar el código con elementos div o span innecesarios.

XML Binario

XML Binario es un formato binario de contenido XML diseñado para reducir el tamaño de transmisión de documentos XML a través de la omisión de elementos y atributos desconocidos, lo que permitirá un uso más efectivo de los datos XML en canales de comunicación de banda estrecha, sin perder así funcionalidad ni información.

XForms

XForms es un nuevo lenguaje de etiquetado para formularios Web, diseñado para ser el sustituto de los formularios HTML tradicionales, y que va a permitir a los desarrolladores distinguir entre el propósito del formulario y su presentación. Esta separación clara entre contenido y presentación ofrece grandes ventajas en términos de reutilización, independencia de dispositivo y accesibilidad.

XHTML

Lenguaje de Etiquetado de Hipertexto Extensible (Extensible HyperText Markup Language). Es una versión más estricta y limpia de HTML que nace ante la limitación de uso de éste con las cada vez más abundantes herramientas basadas en XML. XHTML extiende HTML 4.0 combinando la sintaxis de HTML, diseñado para mostrar datos, con la de XML, diseñado para describir los datos.

XLink

Lenguaje de Enlace XML (XML Linking Language). Es un lenguaje que, entre otras funcionalidades, permite crear vínculos bidireccionales entre recursos, lo que implica la posibilidad de moverse en dos direcciones,

facilitando así la obtención de información remota como recursos en lugar de simplemente como páginas Web.

XML

Lenguaje de Etiquetado Extensible (eXtensible Markup Language). Es un lenguaje con una importante función en el proceso de intercambio, estructuración y envío de datos en la Web. Describe los datos de tal manera que es posible estructurarlos utilizando para ello etiquetas, como lo hace HTML, pero que no están predefinidas, delimitando de esta manera los datos, a la vez que favoreciendo la interoperabilidad de los mismos.

XML Base

XML Base permite a los desarrolladores especificar un URI para resolver URIs relativos en vínculos a imágenes externas, applets, programas de procesamiento de formato, hojas de estilo, etc.

XML Encryption

Lenguaje de Encriptación XML (XML Encryption). Es un lenguaje cuya función principal es asegurar la confidencialidad de partes de documentos XML a través de la encriptación parcial del documento. Encriptación XML se puede aplicar a cualquier recurso Web, incluyendo contenido que no es XML.

XML Key Management

Gestión de Claves XML (XML Key Management - XKMS). Es un protocolo para distribuir y registrar claves públicas que oculta la complejidad que surge con PKI (Infraestructura de Clave Pública). Está compuesto de dos partes que son: el registro de la clave pública (X-KRSS) y la información de clave pública (X-KISS)

XML Query (XQuery)

Lenguaje de Consulta XML (XML Query). Es un lenguaje que ofrece la posibilidad de realizar consultas en infinidad de tipos diferentes de documentos como son documentos estructurados, colecciones de documentos, bases de datos, estructuras DOM, catálogos, etc., para extraer datos en la Web.

XML Schema

Esquema XML (XML Schema). Es un lenguaje cuyo objetivo principal es definir la estructura en bloques de un documento XML, al igual que lo hace un DTD, pero de una forma mucho más precisa. El propósito de un esquema es definir y describir una clase de documentos XML usando estas construcciones para restringir y documentar el significado, uso y

relaciones de las partes constituidas: tipo de datos, elementos y su contenido, atributos y sus valores, entidades y su contenido, y anotaciones. Los esquemas documentan su propio significado, uso y función.

XML Signature

Firma XML (XML Signature). Es un sistema que, a través de una firma digital, permite ofrecer autenticación de los datos que se manejan. Con la firma digital se confirma la identidad del emisor, la autenticidad del mensaje y su integridad, sin olvidar que los mensajes no serán repudiados.

XPath

Lenguaje de Rutas XML (XML Path Language). Es un lenguaje para acceder a partes de un documento XML, como pueden ser sus atributos, elementos, etc.

XPointer

Lenguaje de Direccionamiento XML (XML Pointer Language). Es un lenguaje que se utiliza para llegar a partes específicas de un documento XML, esto es, a sus elementos, atributos y contenido. XPointer, una vez que XLink ha establecido el enlace con el recurso, va a un punto concreto del documento. Su funcionamiento es muy similar al de los identificadores de fragmentos en un documento HTML ya que se añade al final de una URI y después lo que hace es encontrar el lugar especificado en el documento XML.

XSL

Lenguaje de Hojas de Estilo Extensible (eXtensible Stylesheet Language). Es un lenguaje para crear hojas de estilo a través de las cuales será posible mostrar el contenido estructurado de un documento con un formato determinado. Consiste en dos partes: un lenguaje de transformación de documentos XML y un vocabulario XML para especificar semánticas para el formato (objetos de formato).

XSLT

Transformaciones del Lenguaje de Hojas de Estilo Extensible (Extensible Stylesheet Language Transformations). Es un lenguaje que permite la transformación de la estructura de un documento XML en otro documento XML con estructura diferente.

b) Existe cierta confusión en la comunidad web sobre la relación de los términos URI, URL y URN (*Uniform Resource Identifier*). Amplia dicha información, profundizando sobre dichos formatos consultando la web <http://www.w3.org/Addressing/> de la W3C o investiga en Internet hasta tener claro dichos conceptos. Elabora un breve documento sobre la relación entre dichos términos.

Lo más común que puede pasar es no saber si poner URL o URI. Estos conceptos no están claros, ya que en varias fuentes dicen cosas distintas. Primero hay que indicar unas definiciones de cada elemento:

- URL: **Uniform Resource Locator**, o Localizador Uniforme de Recursos. Son unas cadenas de texto que se usan para nombrar recursos en Internet para su localización.
Ejemplo:
`http://es.wikipedia.org:80/wiki/Special:Search?search=tren&go=Go`
- URN: **Uniform Resource Name**, o Nombre Uniforme de Recursos. Son unas cadenas de texto que se usan para nombrar recursos en Internet para su identificación.
Ejemplo: `urn:isbn:0451450523`
- URI: **Uniform Resource Identifier**, o Identificador Uniforme de Recursos.

Las **URI engloban los otros dos conceptos**, pudiendo clasificarlas entre URL, URN o ambas. En el esquema de arriba se puede observar el contenido del conjunto URI, y la intersección entre los conjuntos URL y URN.

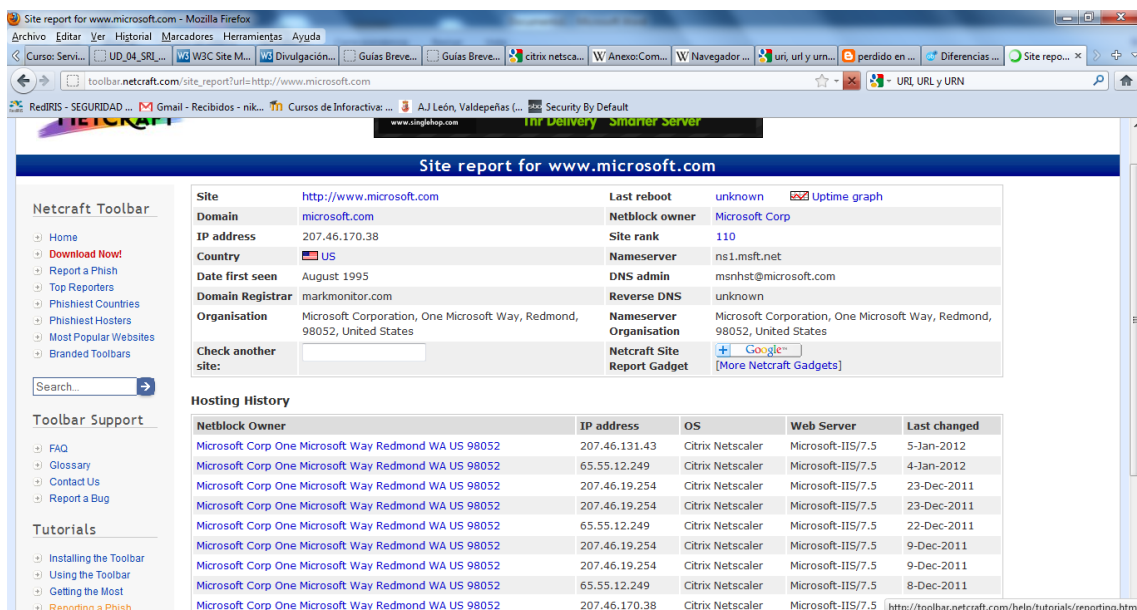
Las **URN** no son utilizadas actualmente. Sirven para identificar pero no para localizar.

c) Visita la web <http://netcraft.com/> que ofrece servicios de análisis y comparativas de la Web. En su enlace “What’s that site running” se permite consultar información sobre el servidor web en el que se ejecuta o aloja un sitio web determinado. Indica en un informe los servidores web que dan soporte a tus sitios web preferidos. Por ejemplo: www.iesgregorioprieto.org, www.jccm.es, ...etc. Incluye asimismo el gráfico correspondiente al ranking de Servidores Web utilizados en Internet.

En primer lugar deberemos de introducir la dirección a buscar, en mi caso www.microsoft.com

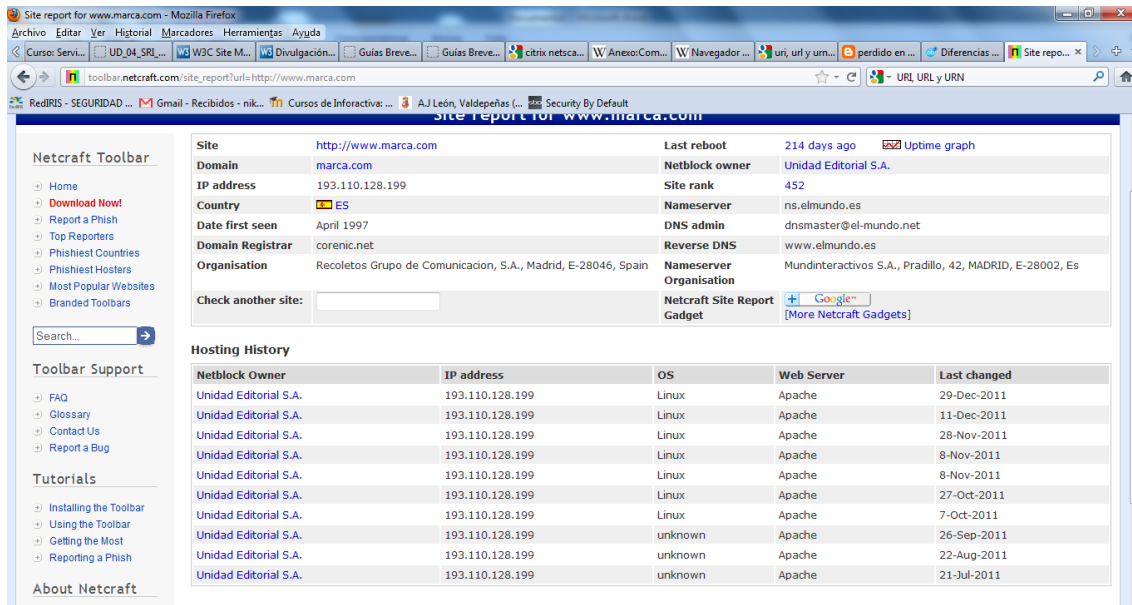


Ahora nos aparecerán los resultados con el SO y el servidor web que usa dicha pagina.



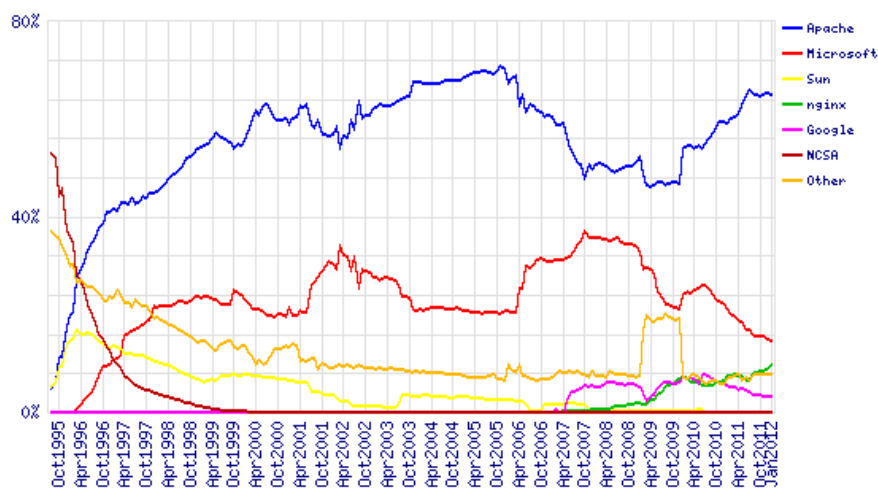
Como conclusión podremos indicar que para la página de www.microsoft.com el servidor Web que utiliza es el Microsoft-IIS/7.5 y el SO que utiliza es el Centrix Netscaler.

Ahora analizaremos la página www.marca.com:



Como resultado podemos indicar que esta página usa como servidor web el APACHE y como SO un Linux.

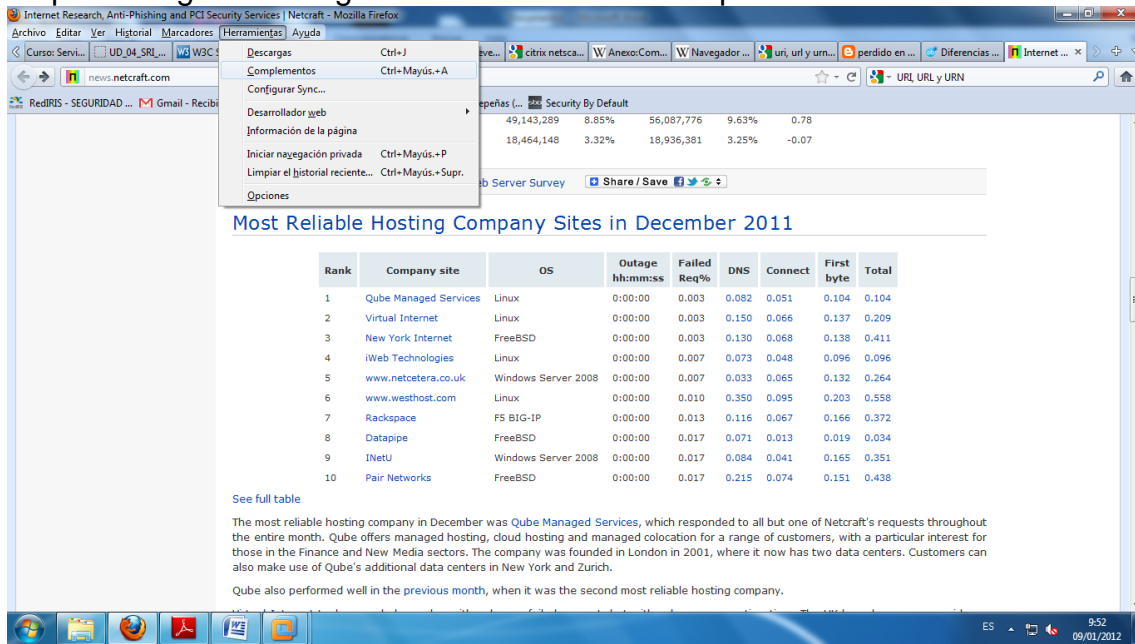
En este gráfico podemos ver el ranking de Servidores Web utilizados en Internet.



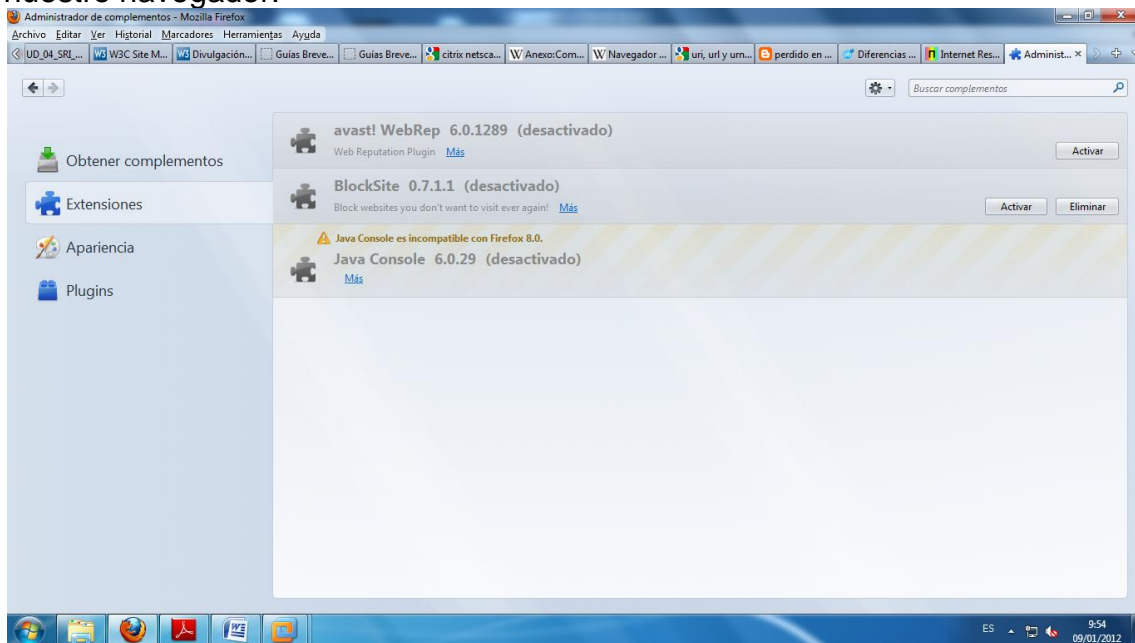
Developer	December 2011	Percent	January 2012	Percent	Change
Apache	362,267,922	65.22%	378,267,399	64.91%	-0.30
Microsoft	82,521,809	14.86%	84,288,985	14.46%	-0.39
nginx	49,143,289	8.85%	56,087,776	9.63%	0.78
Google	18,464,148	3.32%	18,936,381	3.25%	-0.07

d) Inicia sesión con el navegador Firefox en una distribución GNU/Linux. Accede en la barra de menú a herramientas, complementos. Observa las extensiones, temas, idiomas y *plugins* instalados. Haz clic en obtener complementos y verás una lista de complementos disponibles para descargar e instalar. Instala al menos uno. Documenta dicho proceso. Si no tienes instalado dicho navegador accede a su web (<http://www.mozilla-europe.org/es/>) e instala el mismo.

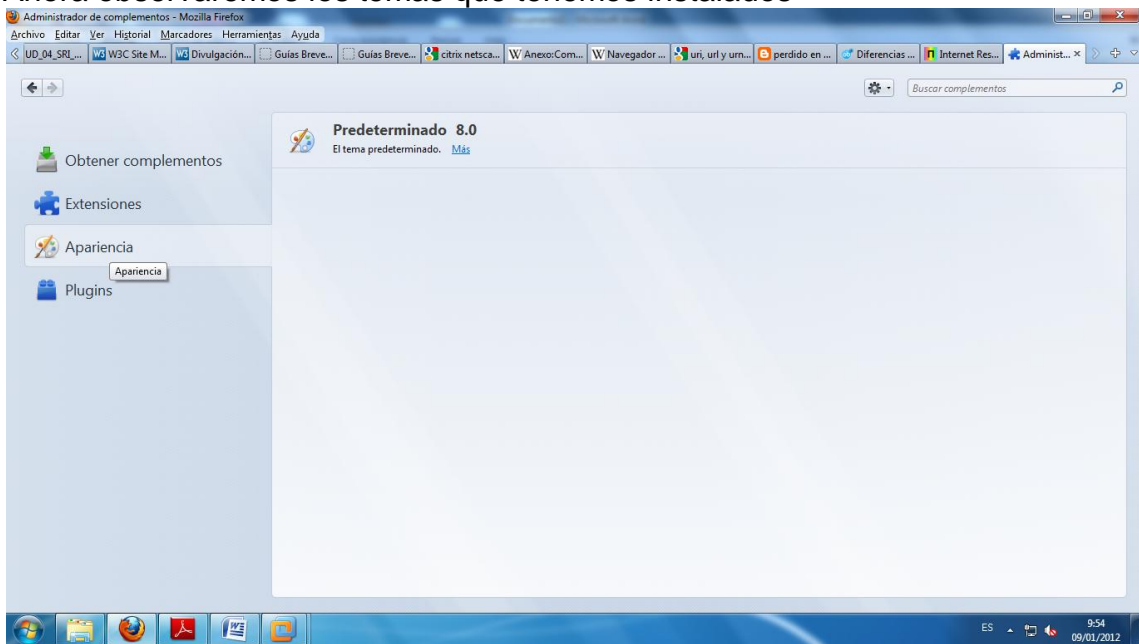
En primer lugar nos dirigimos a herramientas/complementos:



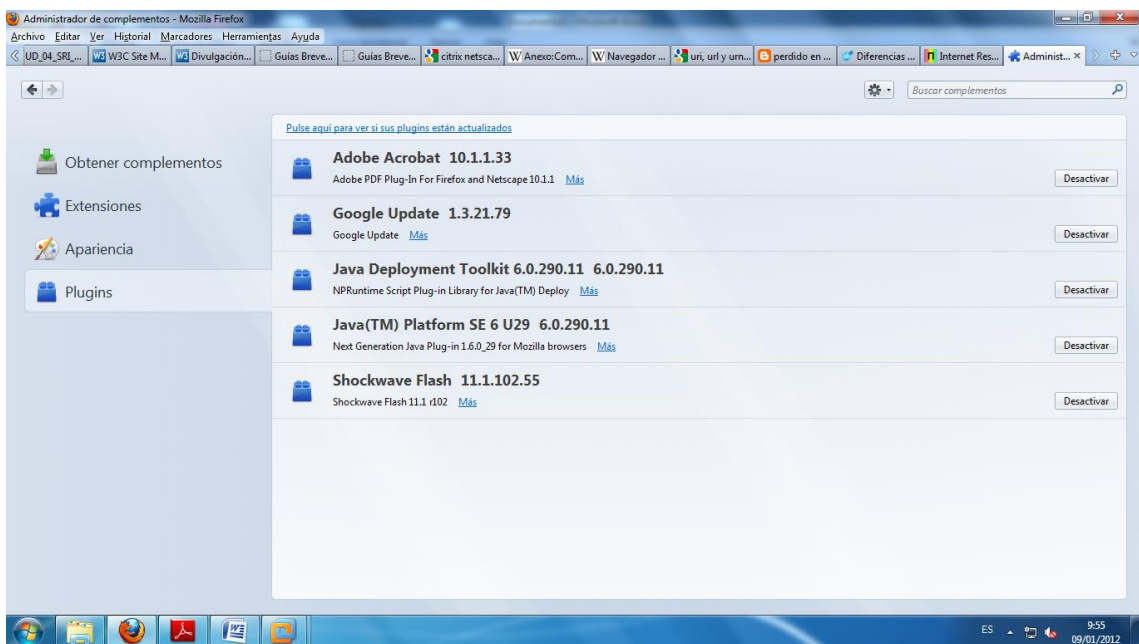
Ahora en primer lugar observaremos las extensiones que tiene instaladas nuestro navegador:



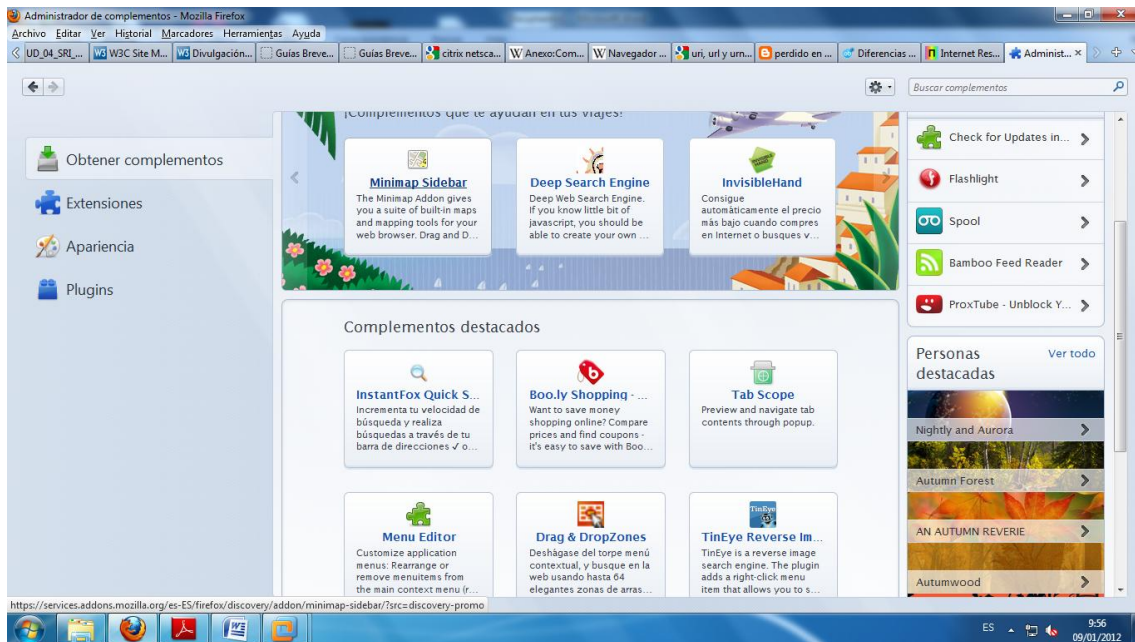
Ahora observaremos los temas que tenemos instalados



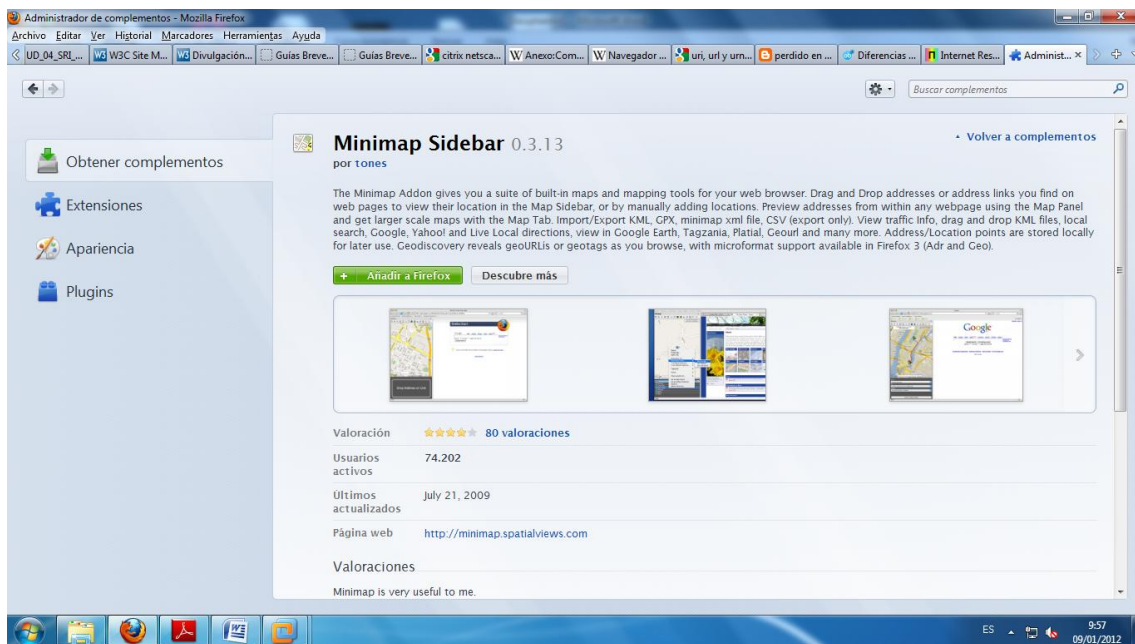
Por último observaremos los plugins que tenemos instalados:



Ahora nos dirigiremos a la opción obtener complementos y procederemos a la instalación de uno:



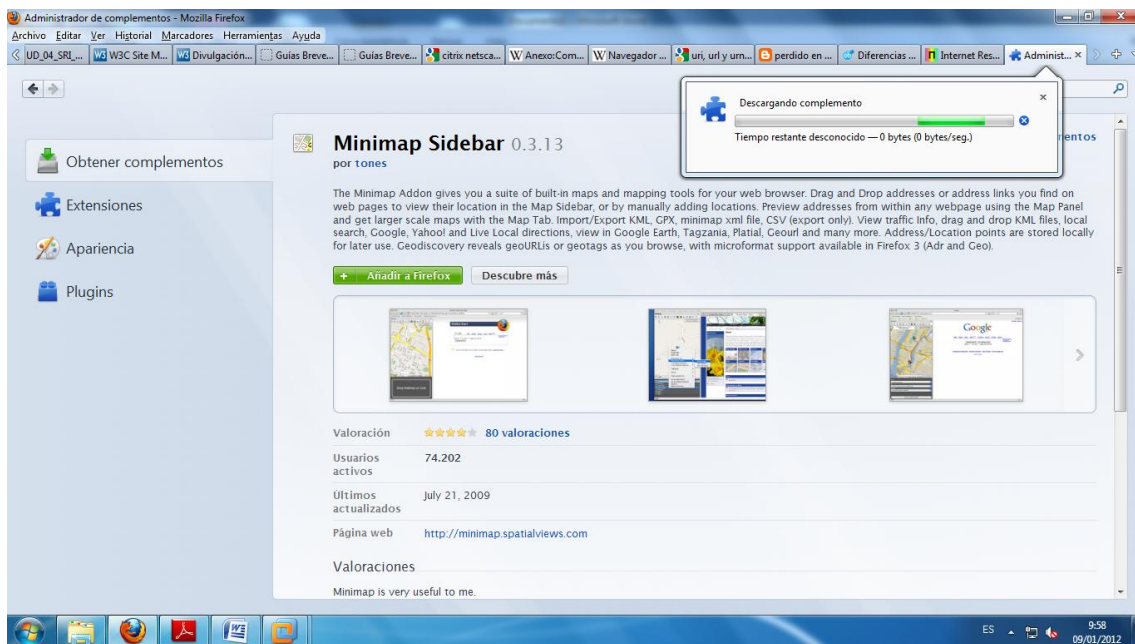
El complemento que instalaremos será el minimap sidebar:



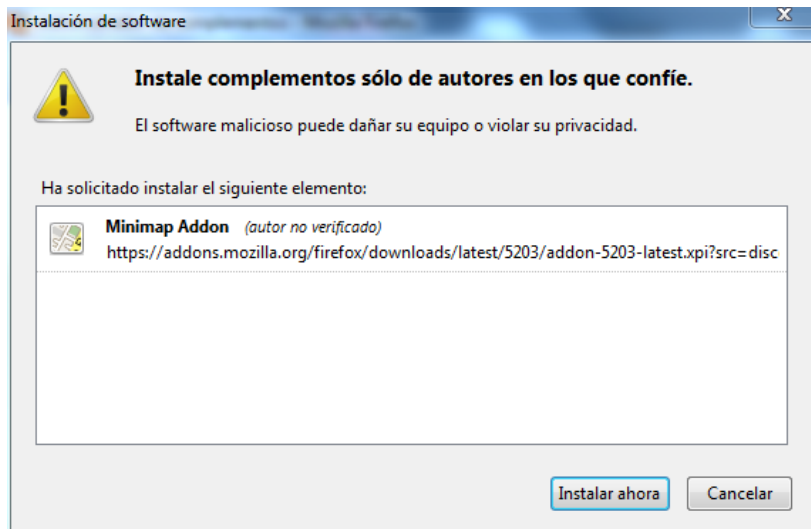
En la pantalla anterior pulsamos sobre añadir a firefox y pulsamos sobre la opción permitir:



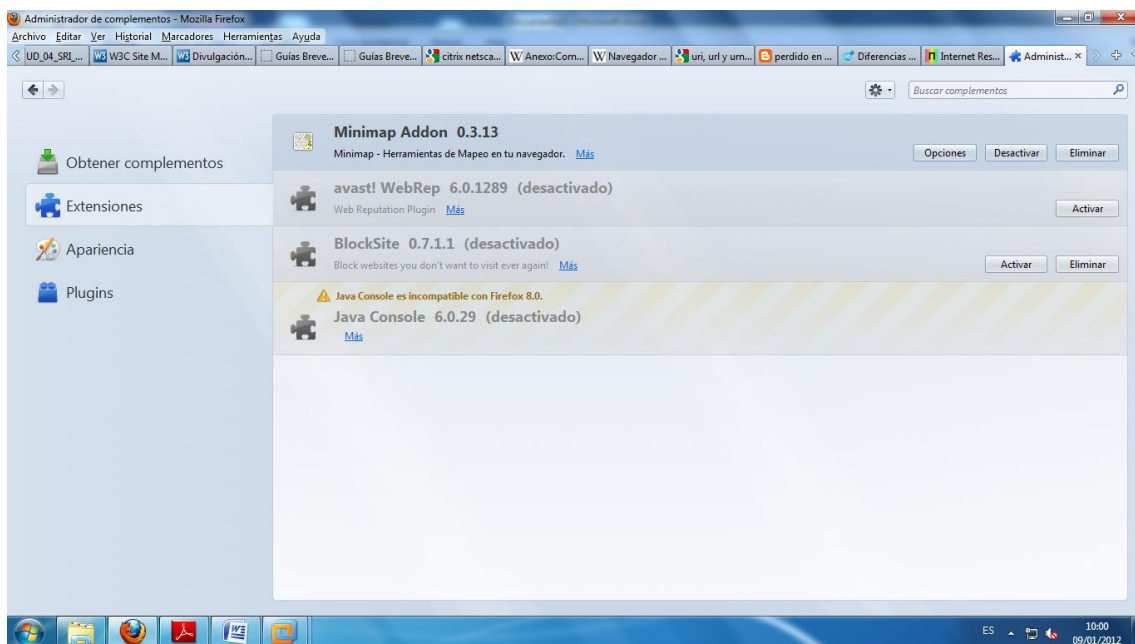
Ahora daré comienzo a la descarga e instalación del complemento deseado:



Ahora pulsamos sobre la opción instalar ahora y reiniciamos el firefox para que los cambios se realicen de manera satisfactoria:



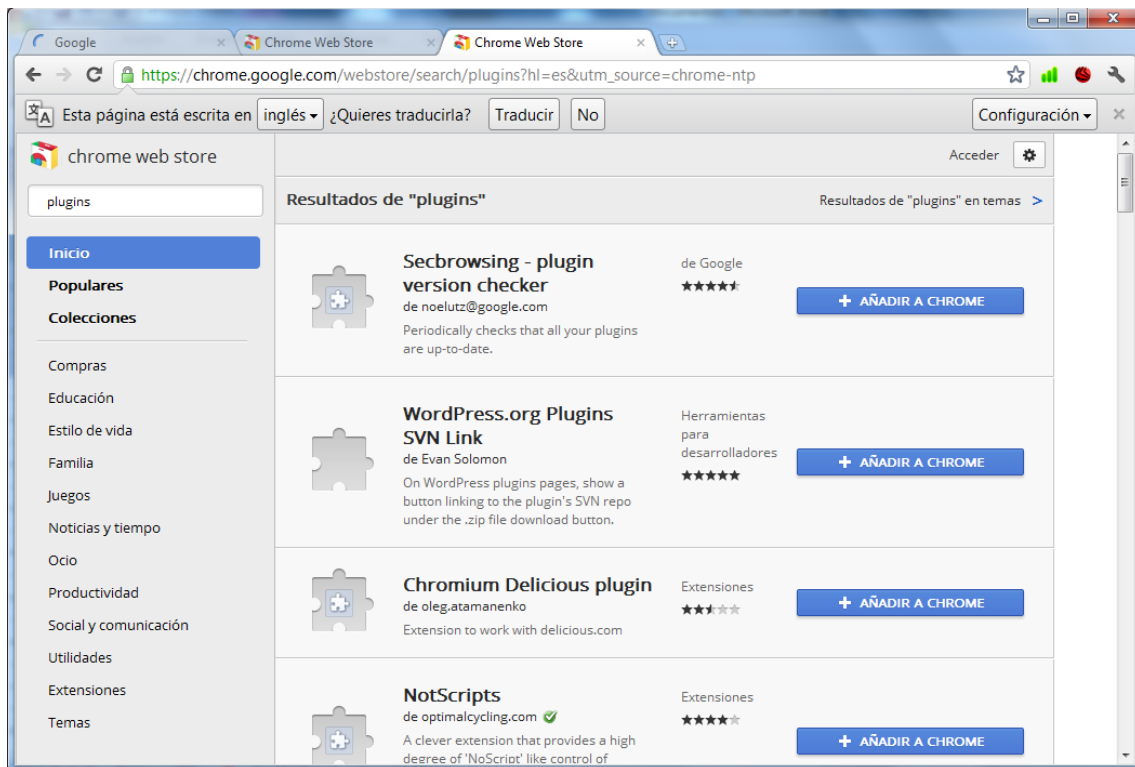
Ahora nos dirigimos a la pestaña complemento y observaremos que nos aparece el complemento instalado:



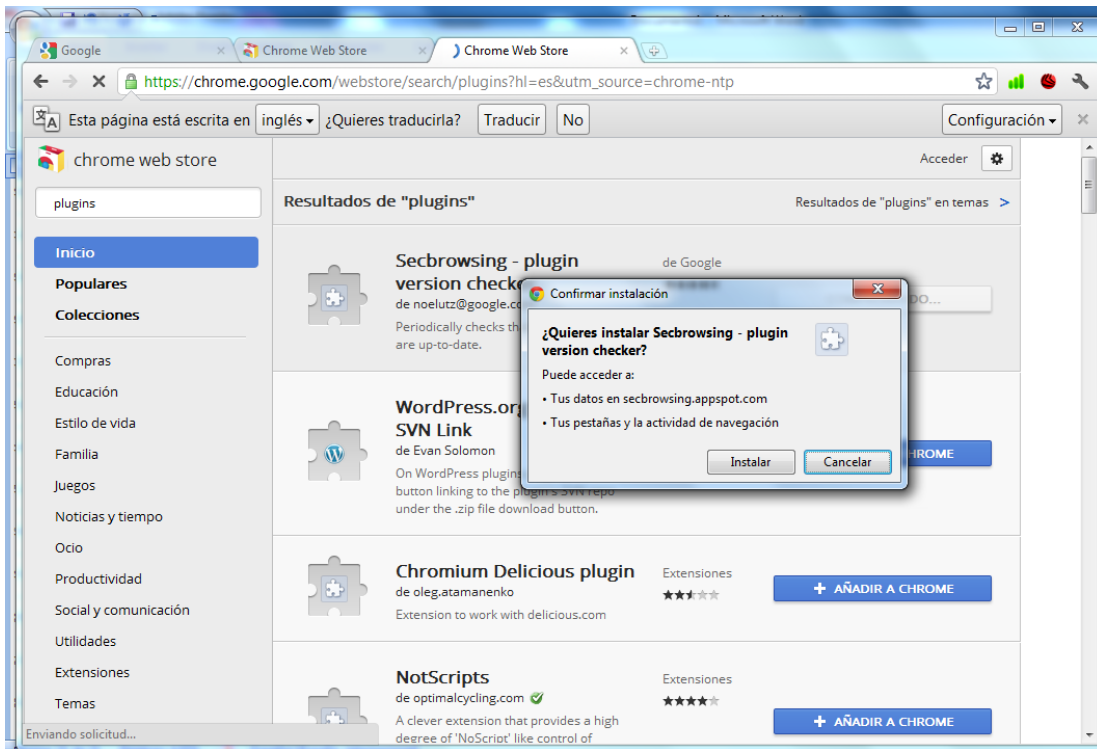
e) Realiza y documenta la instalación de *plugins* para los navegadores Internet Explorer (instalado por defecto en Windows) y Google Chrome (<http://www.google.es/chrome>). Si necesitas instalar dichos navegadores visita sus sitios web de descarga.

Instalación de plugins en chrome:

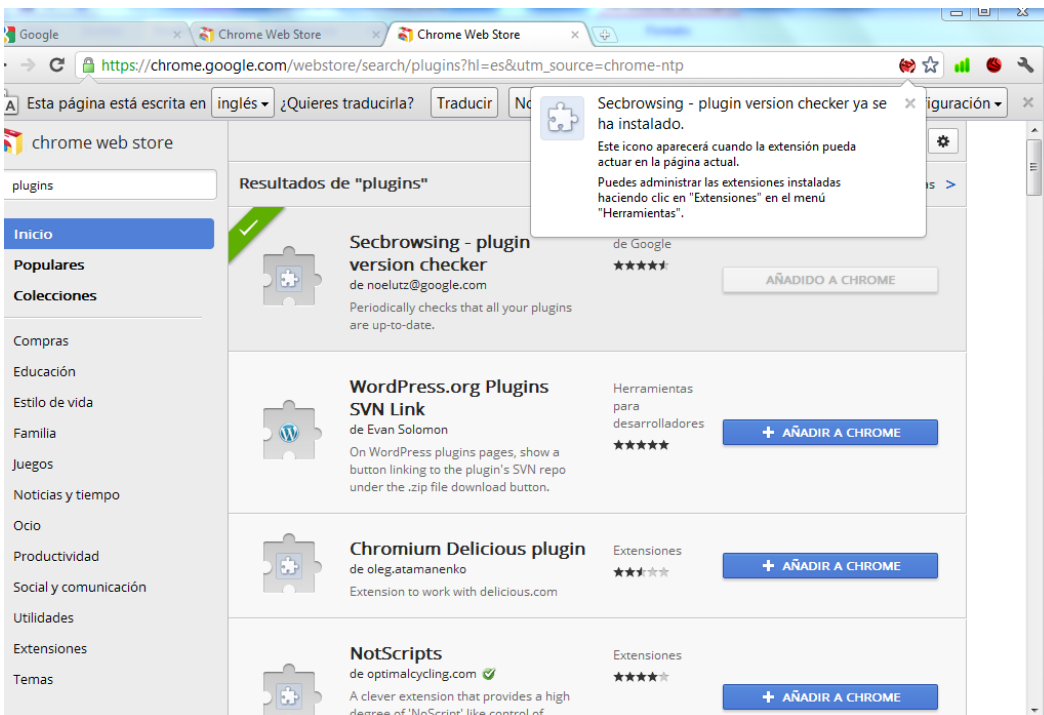
En primer lugar nos dirigimos a la dirección https://chrome.google.com/webstore?hl=es&utm_source=chrome-ntp, una vez allí en la casilla de búsqueda ponemos plugins:



Ahora instalaremos el plugin Secbrowsing, para ello pulsamos sobre añadir a chrome y confirmamos la instalación:



Una vez realizado el paso anterior habremos instalado un plugin:

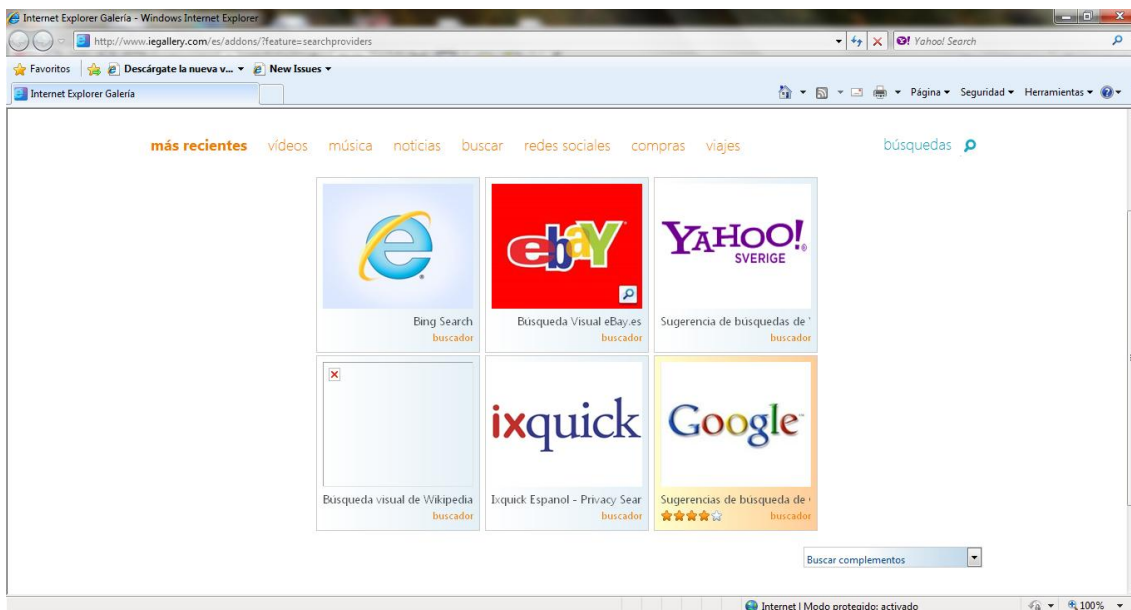


En internet explorer

En primer lugar nos dirigimos a la pagina web <http://www.iegallery.com/es/> y nos metemos en la pestaña complementos:



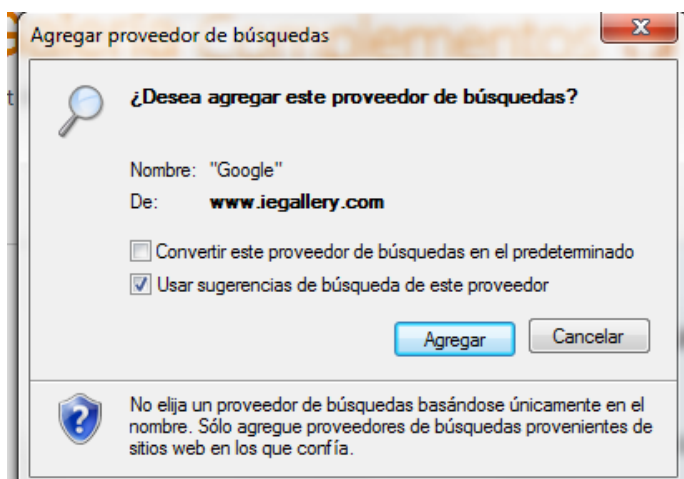
En nuestro lugar instalaremos el complemento sugerencias de búsqueda de google, para ello pulsamos sobre el icono



Ahora pulsamos en la opción agregar a internet explorer:



Ahora pulsamos en agregar:

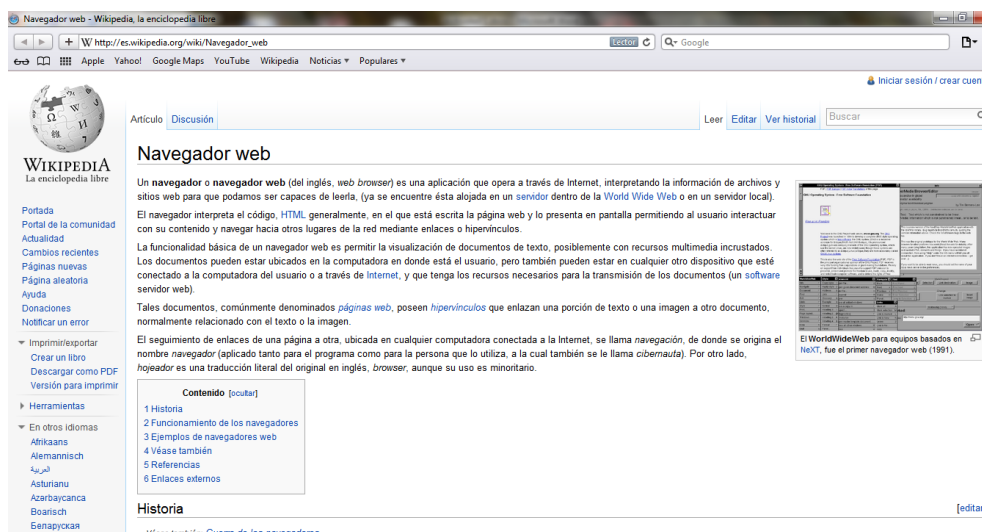


Ahora podremos apreciar que nos aparecen las sugerencias de google:

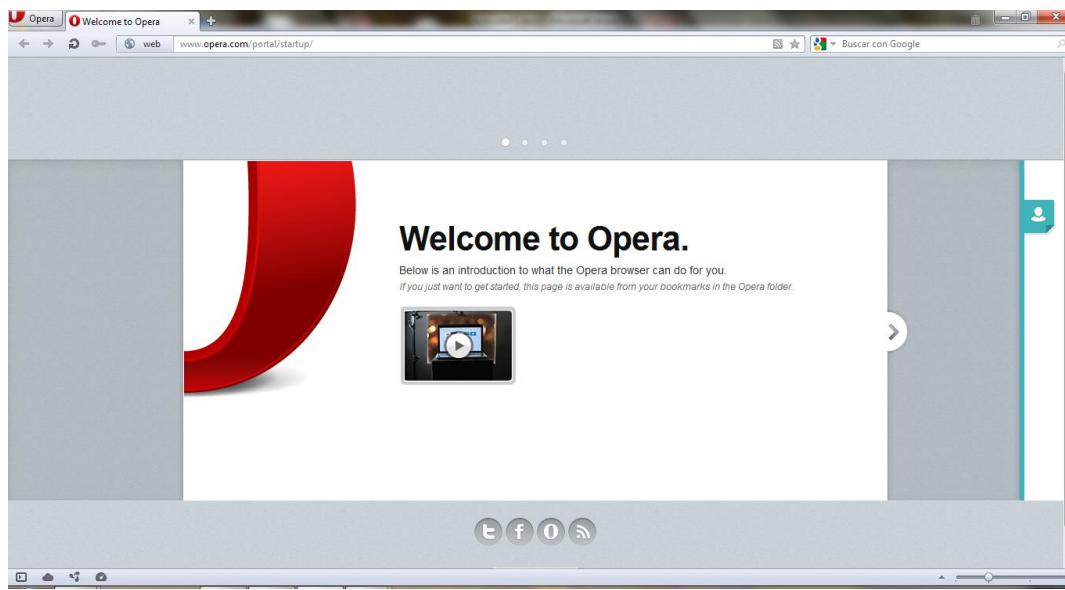


f) Instala los navegadores Safari (<http://www.apple.com/es/safari/>) Opera (<http://www.opera.com>), Amaya (<http://www.w3.org/Amaya/>) y el navegador de texto Lynx (<http://lynx.isc.org/>). Documenta con ayuda de la Wikipedia la historia de dichos navegadores web, sus características y compara los mismos. Comienza consultando el sitio web: http://es.wikipedia.org/wiki/Navegador_web y http://es.wikipedia.org/wiki/Anexo:Tabla_comparativa_de_navegadores_web. Indica cuál es para ti el mejor de todos los navegadores web existentes en el mercado informático. Razona tu respuesta.

SAFARI



OPERA



LYNX

Una vez instalado para ejecutarlo introducimos el siguiente comando:

```
root@ubuntu1:/home/niko# lynx
```

En el siguiente ejemplo podemos ver como se ha accedido a la página web del marca a través de dicho navegador:

```
root@ubuntu1:/home/niko
Archivo Editar Ver Buscar Terminal Ayuda
<<<< MARCA.com (p1 of 51)
REFRESH(900 sec): http://www.marca.com/
#RSS MARCA.com (Portada)

Hoy en MARCA: La Liga, más blanca
* Vivienda
* Empleo
* Coches

* Radio MARCA
* MARCA TV
* MARCA Sports Café
* MARCAEntradas
* La Tienda de MARCA

MARCA.com

Lideres en internet: ayer, 3.163.144 usuarios
Trofeo Pichichi-Zamora

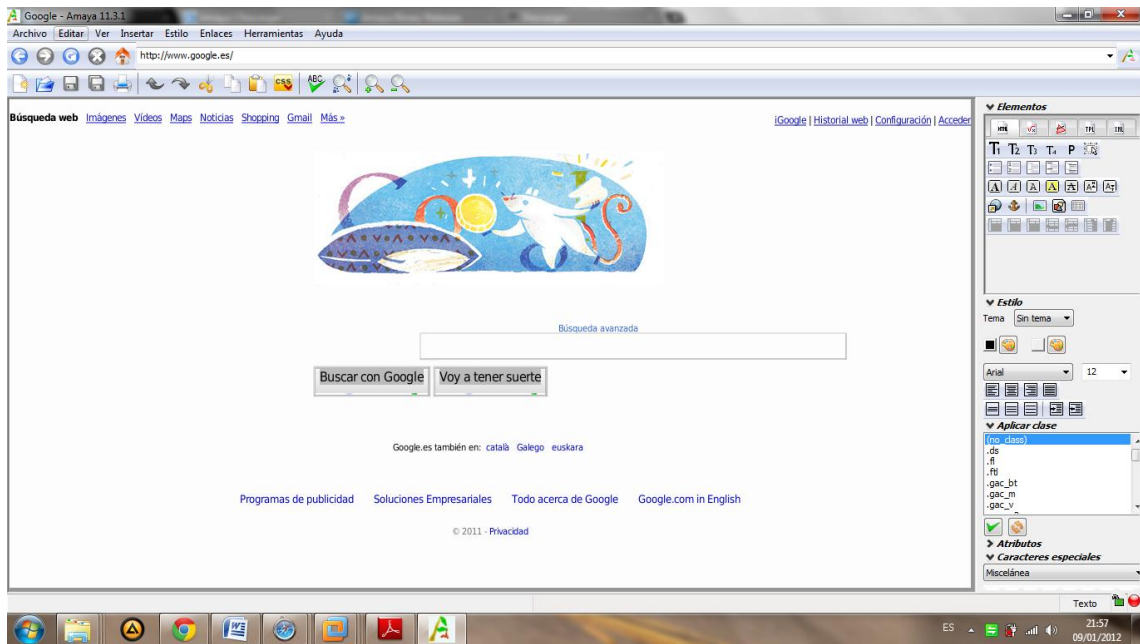
Pichichi y Zamora

Los goleadores y los mejores porteros de Primera y Segunda
El golazo

Elige el golazo

Vota por el mejor
tanto de la jornada
Trofeo Di Stefano
Trofeo Di Stefano
```

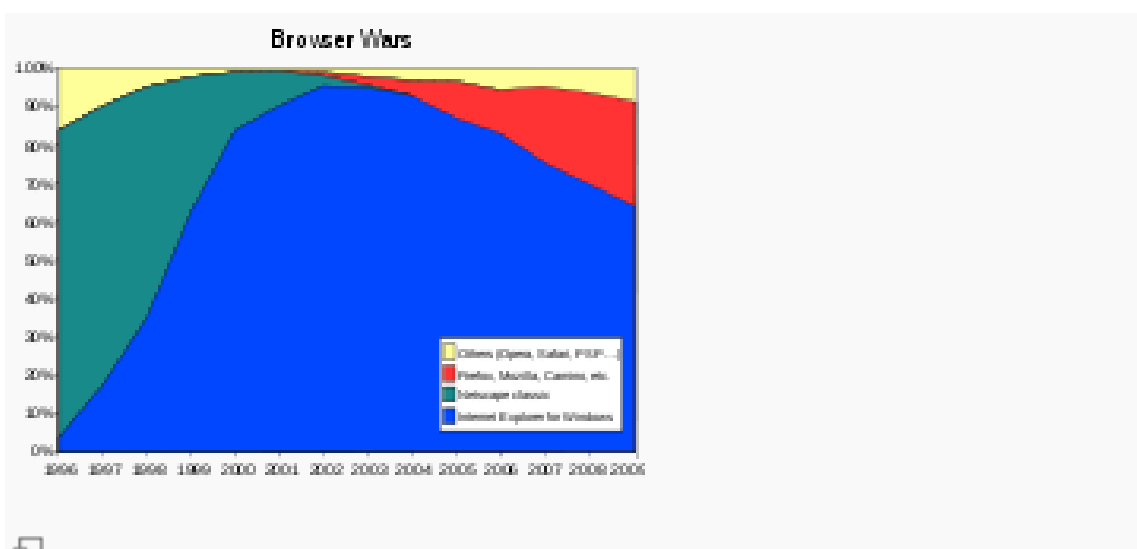
AMAYA



Historia

El primer navegador, desarrollado en el CERN a finales de 1990 y principios de 1991 por Tim Berners-Lee, era bastante sofisticado y gráfico, pero sólo funcionaba en estaciones NeXT.

El navegador **Mosaic**, que funcionaba inicialmente en entornos UNIX sobre X11, fue el primero que se extendió debido a que pronto el NCSA preparó versiones para Windows y Macintosh. Sin embargo, poco más tarde entró en el mercado **Netscape Navigator** que rápidamente superó en capacidades y velocidad a Mosaic. Este navegador tuvo la ventaja de funcionar en casi todos los UNIX, así como en entornos Windows.



Estimación del uso en el período 1996-2006.

Internet Explorer (anteriormente Spyglass Mosaic) fue la apuesta tardía de Microsoft para entrar en el mercado y consiguió desbancar al Netscape Navigator entre los usuarios de Windows, debido a la integración del navegador con el sistema operativo, llegando a poseer cerca del 95% de la cuota de mercado. Netscape Communications Corporation liberó el código fuente de su navegador, naciendo así el proyecto **Mozilla**.

Finalmente **Mozilla Firefox** fue reescrito desde cero tras decidirse a desarrollar y usar como base un nuevo conjunto de widgets multiplataforma basado en XML llamado XUL y esto hizo que tardara bastante más en aparecer de lo previsto inicialmente, apareciendo una versión 1.0 de gran calidad y para muchísimas plataformas a la vez el 5 de junio del 2002.

A finales de 2004 aparece en el mercado **Firefox**, una rama de desarrollo de Mozilla que pretende hacerse con parte del mercado de Internet Explorer. Se trata de un navegador más ligero que su hermano mayor.

El 2 de septiembre del 2008 **Google Chrome** vio la luz. Es el navegador web desarrollado por Google y compilado con base en componentes de código abierto como el motor de renderizado de WebKit y su estructura de desarrollo de aplicaciones (Framework). Google Chrome es el tercer navegador más utilizado en Internet y actualmente posee una cuota de mercado del 7,05%.¹ Está disponible gratuitamente bajo condiciones de servicio específicas. El nombre del navegador deriva del término usado para el marco de la interfaz gráfica de usuario ("chrome").

Comparativa de los navegadores:

Comparativa de los ordenadores respecto a su compatibilidad con SO

Ordenes operativos en los que los navegadores funcionan sin limitación.

	Windows	Mac OS X	Mac OS 9	GNU/Linux	BSD	Unix	GNX	GNU Hurd
Amaya	✓ si	✓ si	✗ no	✓ si	✓ si	✓ si	✗ no	✓ si
Camino	✗ no	✓ si	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no
Chrome	✓ si	✓ si	✗ no	✓ si	✗ no	✗ no	✗ no	✗ no
Dillo	Parcial	✓ si	✗ no	✓ si	✓ si	✓ si	✗ no	✓ si
ELinks	✓ si	✓ si	✗ no	✓ si	✓ si	✓ si	✗ no	✓ si
Epiphany	✗ no	✓ si	✗ no	✓ si	✓ si	✓ si	✗ no	✗ no
Galeon	✗ no	✓ si	✗ no	✓ si	✓ si	✓ si	✗ no	✗ no
iCab	✗ no	✓ si	✓ si	✗ no	✗ no	✗ no	✗ no	✗ no
Internet Explorer	✓ si	✗ no	✗ no	✗ no	✗ no	Abandonado (5.0)	✗ no	✗ no
Internet Explorer para Mac	✗ no	Terminado (5.2.3)	Abandonado (5.1.7)	✗ no	✗ no	✗ no	✗ no	✗ no
K-Meleon	✓ si	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no
Konqueror	✓ si	✓ si	✗ no	✓ si	✓ si	✓ si	✗ no	✗ no
Links	Parcial	✓ si	✗ no	✓ si	✓ si	✓ si	✗ no	✓ si
Lynx	✓ si	✓ si	✓ si	✓ si	✓ si	✓ si	✗ no	✓ si
Midori	✓ si	✓ si	✗ no	✓ si	✓ si	✓ si	✓ si	✓ si
Mosaic	Terminado (3.0)	✗ no	Terminado (3.0)	Abandonado (2.6)	Abandonado (2.6)	Abandonado (2.6)	✗ no	✗ no
Mozilla	Terminado (1.7.13)	Terminado (1.7.13)	Abandonado (1.2.1)	Terminado (1.7.13)	Terminado (1.7.13)	Terminado (1.7.13)	✗ no	✗ no
Mozilla Firefox	✓ si	✓ si	✗ no	✓ si	✓ si	✓ si	✗ no	✓ si (loewease)
Netscape	✓ si	✓ si	Abandonado	✓ si	✓ si	✓ si	✗ no	✗ no
OmniWeb	✗ no	✓ si	✗ no	✗ no	✗ no	✗ no	Abandonado (6)	✗ no
Opera	✓ si	✓ si	Abandonado (6.03)	✓ si	✓ si	✓ si	✗ no	✗ no
Safari	✓ si	Incluido	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no
SeaMonkey	✓ si	✓ si	✗ no	✓ si	✓ si	✓ si	✗ no	✗ no
WorldWideWeb (NeXTSTEP sólo)	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no
	Windows	Mac OS X	Mac OS 9	GNU/Linux	BSD	Unix	GNX	GNU Hurd

Compatibilidad del navegador

Comparativa de los ordenadores respecto a las características de los navegadores

Información sobre que características o utilidades son implementadas nativamente (sin necesidad de terceros).

	Administrador de marcadores	Gestor de descargas	Cliente de BitTorrent	Administrador de contraseñas	Administrador de formularios	Corrección ortográfica	Barra de herramientas de motor de búsqueda
Amaya	✓ sí	✗ no	✗ no	✗ no	✗ no	✓ sí	✗ no
Camino	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✓ sí	✓ sí
Chrome	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✓ sí	✓ sí
Dillo	Parcial	✓ sí	✗ no	✗ no	✗ no	✗ no	Parcial
ELinks	✓ sí	✓ sí	✗ no	✓ sí	?	✗ no	✗ no
Epiphany	✓ sí	✓ sí	✗ no	✓ sí	✗ no	✓ sí	✓ sí
Galeon	✓ sí	✓ sí	✗ no	✓ sí	✗ no	✗ no	✓ sí
iCab	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✗ no	✓ sí
Internet Explorer ¹	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✗ no ²	✓ sí ³
Internet Explorer para Mac	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✗ no	✗ no
K-Meleon	✓ sí	✗ no	✗ no	✓ sí	✓ sí	SpellCheck KPlugin	✓ sí
Konqueror	✓ sí	Gestor KDE (Kget)	Gestor KDE (Ktorrent)	Gestor KDE (KWallet)	Gestor KDE (KWallet)	✓ sí	✓ sí
Links	✓ sí	✓ sí	✗ no	✗ no	✗ no	✗ no	✗ no
Lynx	✓ sí	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no
Midori	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✓ sí	✓ sí
Mosaic	✓ sí	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no
Mozilla	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✓ sí	✓ sí
Mozilla Firefox	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✓ sí	✓ sí
Netscape	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✓ sí	✓ sí
OmniWeb	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✓ sí	✓ sí
Opera	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí
Safari	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✓ sí	✓ sí
SeaMonkey	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✓ sí	✓ sí
WorldWideWeb	✓ sí	✗ no	✗ no	✗ no	✗ no	✓ sí	✗ no
	Administrador de marcadores	Gestor de descargas	Cliente de BitTorrent	Administrador de contraseñas	Administrador de formularios	Corrección ortográfica	Barra de herramientas de motor de búsqueda

Comparativa de los ordenadores respecto a su accesibilidad:

	Navegación por pestañas	Anti pop-up	Búsqueda incremental	Bloqueo de publicidad	Zoom de página ⁴	Tecla de acceso HTML
Amaya	✓ sí	N/D	?	?	✓ sí	?
Camino	✓ sí	✓ sí	✓ sí	✓ sí	✗ no	✓ sí
Chrome	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí
Dillo	✓ sí	✗ no	✓ sí	✗ no	✗ no	✗ no
ELinks	✓ sí	✗ no	✓ sí	N/D	N/D	?
Epiphany	✓ sí	✓ sí	✓ sí	✗ no	✓ sí	?
Flock	✓ sí	✓ sí	✓ sí	?	✓ sí	?
Galeon	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	?
iCab	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✓ sí
Internet Explorer	✓ sí	✓ sí	✓ sí	✗ no	✓ sí	✓ sí
Internet Explorer para Mac	✗ no	✗ no	✗ no	✗ no	✗ no	✓ sí
K-Meleon	✓ sí	✓ sí	✓ sí	Parcial	Parcial	✓ sí
Konqueror	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí
Links	✗ no	✗ no	?	N/D	N/D	?
Lynx	✗ no	N/D	?	N/D	N/D	?
Midori	✓ sí	✗ no	✓ sí	✓ sí	✓ sí	✗ no
Mosaic	✗ no	N/D	?	✗ no	?	?
Mozilla	✓ sí	✓ sí	✓ sí	Parcial ⁵	✗ no ⁶	✓ sí
Mozilla Firefox	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí ⁶	✓ sí
Netscape	✓ sí	✓ sí	✓ sí	Parcial ⁵	✗ no	✓ sí
OmniWeb	✓ sí	✓ sí	?	✓ sí	✗ no	?
Opera	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí
Safari	✓ sí	✓ sí	✗ no	✓ sí	✓ sí	✓ sí
SeaMonkey	✓ sí	✓ sí	✓ sí	Parcial ⁶	✗ no ⁶	✓ sí
WorldWideWeb	✗ no	✗ no	✗ no	✗ no	✗ no	?

Comparativa de los ordenadores respecto al soporte de tecnologías:

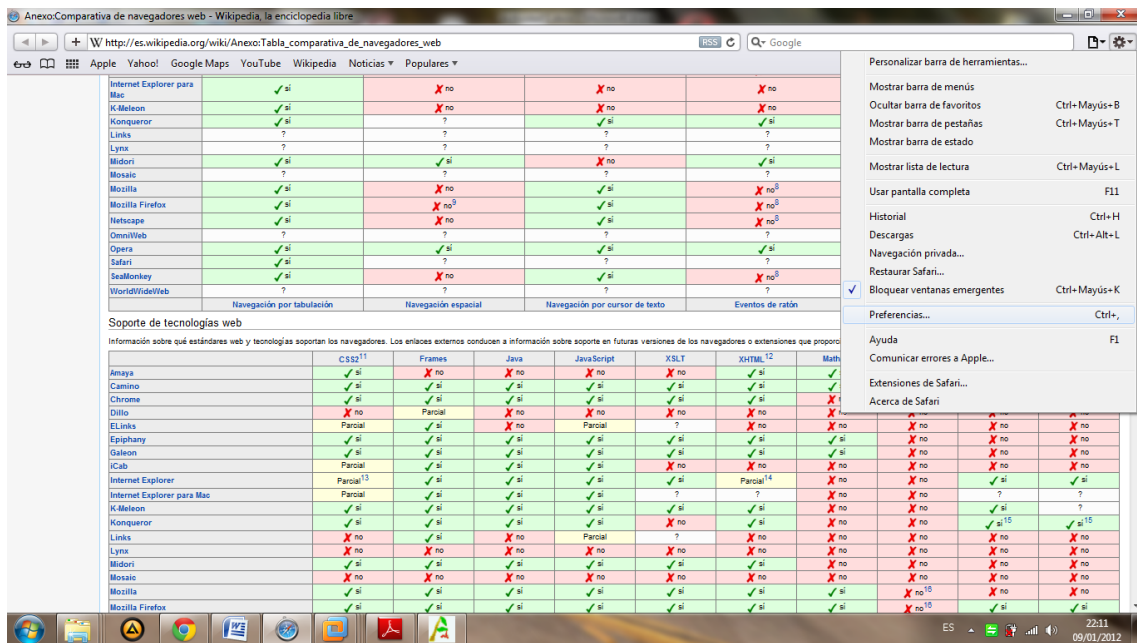
	CSS2 ¹¹	Frames	Java	JavaScript	XSLT	XHTML ¹²	MathML	XForms	RSS	Atom
Amaya	✓ sí	✗ no	✗ no	✗ no	✗ no	✓ sí	✓ sí	✗ no	✗ no	✗ no
Camino	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no	Parcial	Parcial
Chrome	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no	✗ no	✗ no	✗ no
Dillo	✗ no	Parcial	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no
ELinks	Parcial	✓ sí	✗ no	Parcial	?	✗ no	✗ no	✗ no	✗ no	✗ no
Epiphany	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no	✗ no	✗ no
Galeon	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no	✗ no	✗ no
iCab	Parcial	✓ sí	✓ sí	✓ sí	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no
Internet Explorer	Parcial ¹³	✓ sí	✓ sí	✓ sí	✓ sí	Parcial ¹⁴	✗ no	✗ no	✓ sí	✓ sí
Internet Explorer para Mac	Parcial	✓ sí	✓ sí	✓ sí	?	?	✗ no	✗ no	?	?
K-Meleon	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no	✗ no	✓ sí	?
Konqueror	✓ sí	✓ sí	✓ sí	✓ sí	✗ no	✓ sí	✗ no	✗ no	✓ sí ¹⁵	✓ sí ¹⁵
Links	✗ no	✓ sí	✗ no	Parcial	?	✗ no	✗ no	✗ no	✗ no	✗ no
Lynx	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no
Midori	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no	✗ no	✗ no	✗ no
Mosaic	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no
Mozilla	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no ¹⁶	✗ no	✗ no
Mozilla Firefox	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no ¹⁶	✓ sí	✓ sí
Netscape	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no	?	?
OmniWeb	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no	✗ no	✓ sí	✓ sí
Opera	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí
Safari	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no	✗ no	✓ sí	✓ sí
SeaMonkey	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✓ sí	✗ no ¹⁶	✗ no	✗ no
WorldWideWeb	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no	✗ no
	CSS2	Frames	Java	JavaScript	XSLT	XHTML	MathML	XForms	RSS	Atom

En mi opinión el mejor navegador de la actualidad es el Google Chrome, puesto que es el que menos recursos del sistema consume para su utilización y es de los más rápidos, además uno de sus puntos fuertes es la gran cantidad de extensiones que se pueden encontrar para este navegador.

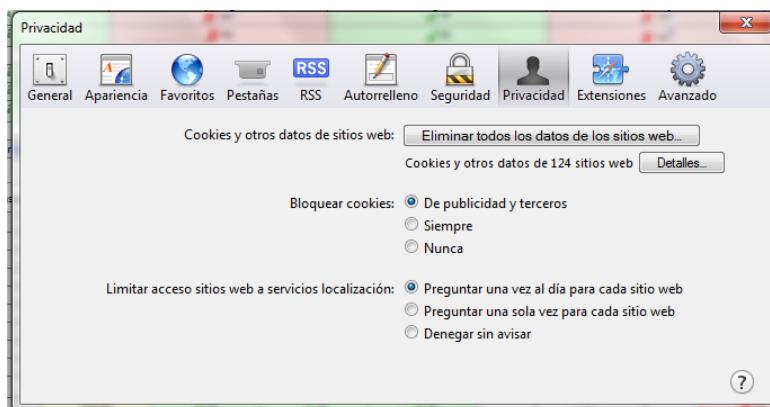
g) Comprueba e investiga en al menos dos navegadores web de los citados anteriormente la configuración de las cookies. Documenta dicho proceso.

Configurar cookies en safari:

En primer lugar nos dirigimos a opciones/ preferencias:

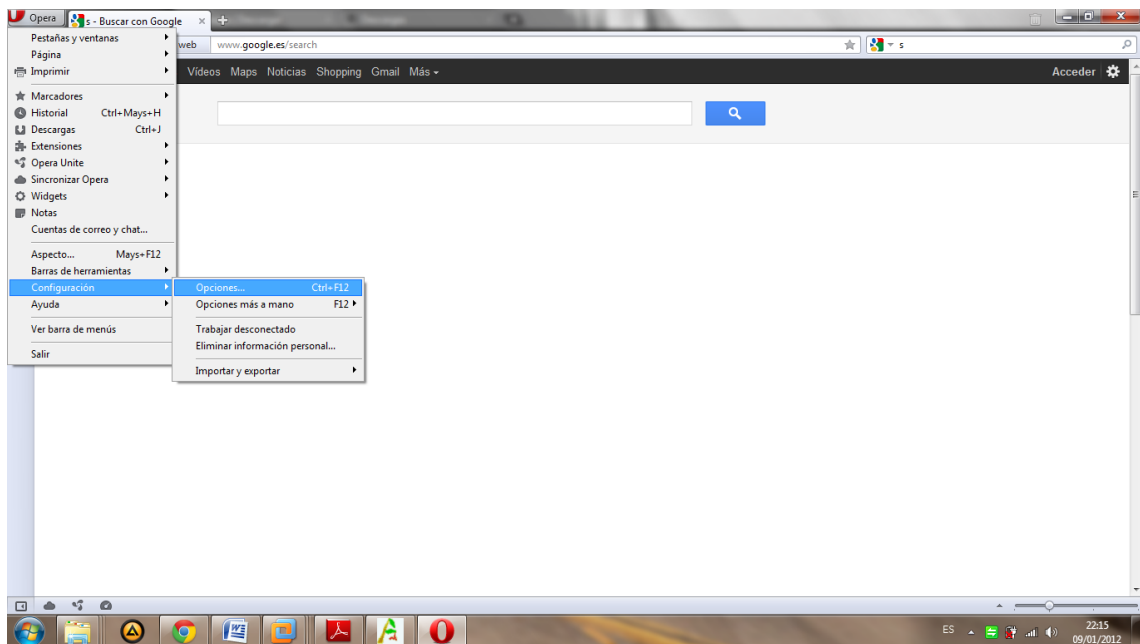


Una vez allí nos dirigimos a la pestaña de privacidad; una vez allí podremos eliminar las cookies o ver los detalles de las mismas:

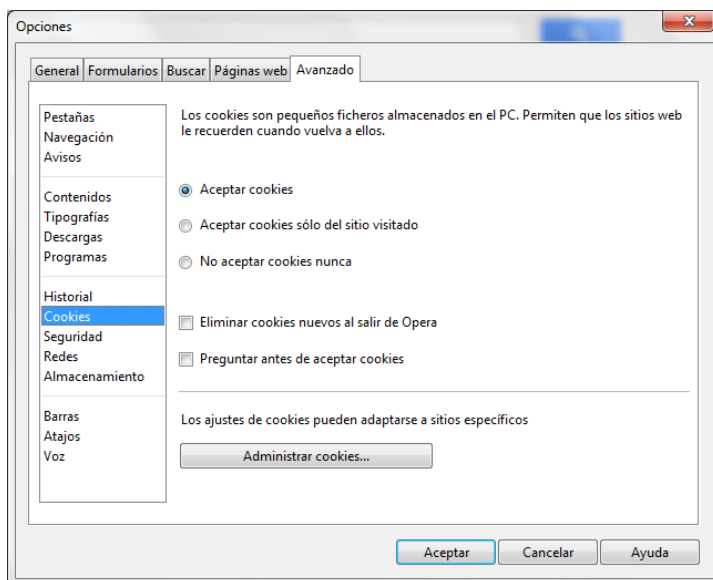


COOKIES EN OPERA:

En primer lugar deberemos de ir a cinfofiguracion/opciones.



Una vez allí nos dirigimos a la pestaña avanzado y una vez allí a la pestaña cookies, aquí podremos aceptar cookies, no aceptar nunca cookies, eliminarlas y administrarlas:



Actividad 2 : Protocolo HTTP. Capturas con sniffer.

```

Stream Content
GET / HTTP/1.1
Host: www.w3c.es
User-Agent: Mozilla/5.0 (windows NT 6.1; WOW64; rv:7.0.1) Gecko/20100101 Firefox/7.0.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-es,es;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Connection: keep-alive

HTTP/1.1 200 OK
Date: Mon, 09 Jun 2012 21:22:11 GMT
Server: Apache
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Transfer-Encoding: chunked
Content-Type: text/html

22ff
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="es" lang="es">

<head>
<title>world wide web Consortium (W3C) - Espa..a</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link rel="Help" href="/ayuda/" />
<!-- Estilos -->
<link rel="stylesheet" href="/css/2010/minimum" type="text/css" media="handheld, all" />
<link rel="stylesheet" href="/css/2010/es" type="text/css" media="handheld, all" />
<style type="text/css" media="print, screen and (min-width: 481px)">
</style>
<!-- [CDATA[
<script src="/css/2010/advanced">
</script>
</style>
<link href="/css/2010/minimum" rel="stylesheet" type="text/css" media="handheld, only screen and (max-device-width: 480px)" />
<meta name="viewport" content="width=320" />
<link rel="stylesheet" href="/css/2010/print" type="text/css" media="print" />
<link rel="shortcut icon" href="/img/2010/favicon.ico" type="image/x-icon" />
<!-- Alternativas y metadatos -->
<meta name="description" content="El world wide web Consortium (W3C) es una comunidad internacional donde las organizaciones Miembro, personal a tiempo completo y el p...dico trabajan para desarrollar est...ndares web." />
<link rel="alternate" type="application/rss+xml" title="Noticias de W3C Oficina Espa..a (formato RSS 1.0)" href="http://www.w3c.es/Noticias/canal/rdf/" />

```

a) ¿Qué versión de HTTP se utiliza? ¿Qué método se ha usado en la primera petición HTTP?

Utiliza la versión 1.1 de http, ha utilizado el método GET.

b) ¿Qué valor tiene la cabecera *Host*?. ¿Para qué las usará el servidor?.

www.w3c.es, las usara para identificar la página web.

c) ¿Qué algoritmos de compresión soporta el navegador?

Soporta Gzip y deflate.

```

GET / HTTP/1.1
Host: www.w3c.es
User-Agent: Mozilla/5.0 (windows NT 6.1; WOW64; rv:7.0.1) Gecko/20100101 Firefox/7.0.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: es-es,es;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Connection: keep-alive

```

d) ¿Se envían *cookies* en la petición HTTP?.

No se envían cookies

e) ¿Qué código de estado tiene la primera respuesta HTTP? ¿Qué servidor web responde?

El código de estado de la primera respuesta es el 200

```

HTTP/1.1 200 OK
Date: Mon, 09 Jan 2012 21:22:11 GMT
Server: Apache
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Transfer-Encoding: chunked
Content-Type: text/html

22ff
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="es" lang="es">

```

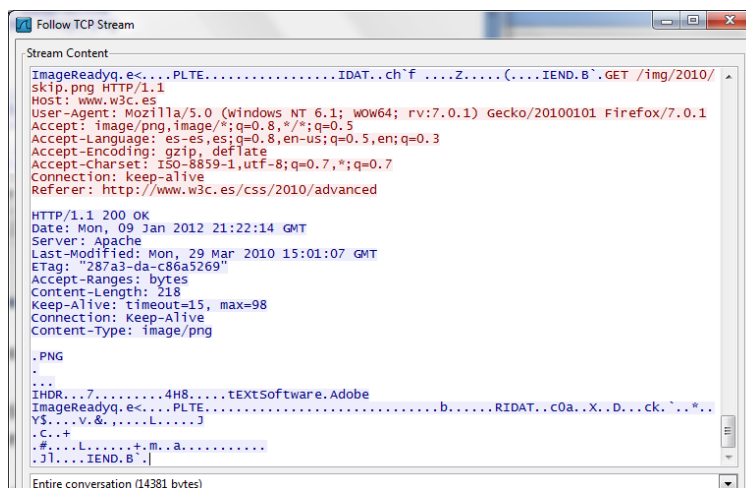
f) ¿De qué tipo MIME es el recurso enviado?.

g) ¿ Se han utilizado conexiones persistentes, es decir, en la misma conexión TCP haya varias peticiones y respuestas HTTP?

Si

h) ¿Existen peticiones y respuestas de imágenes? ¿Por qué lo sabes?.

Si existen, lo se porque se han capturados dichos paquetes:



```

Follow TCP Stream
Stream Content
ImageReadyq.e<...PLTE.....IDAT..ch`f....Z.....(....IEND.B`.GET /img/2010/
skip.png HTTP/1.1
Host: www.w3c.es
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:7.0.1) Gecko/20100101 Firefox/7.0.1
Accept: image/png,image/*;q=0.8,*/*;q=0.5
Accept-Language: es-es,es;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Connection: keep-alive
Referer: http://www.w3c.es/css/2010/advanced

HTTP/1.1 200 OK
Date: Mon, 09 Jan 2012 21:22:14 GMT
Server: Apache
Last-Modified: Mon, 29 Mar 2010 15:01:07 GMT
ETag: "287a3-da-c86a5269"
Accept-Ranges: bytes
Content-Length: 218
Keep-Alive: timeout=15, max=98
Connection: Keep-Alive
Content-Type: image/png

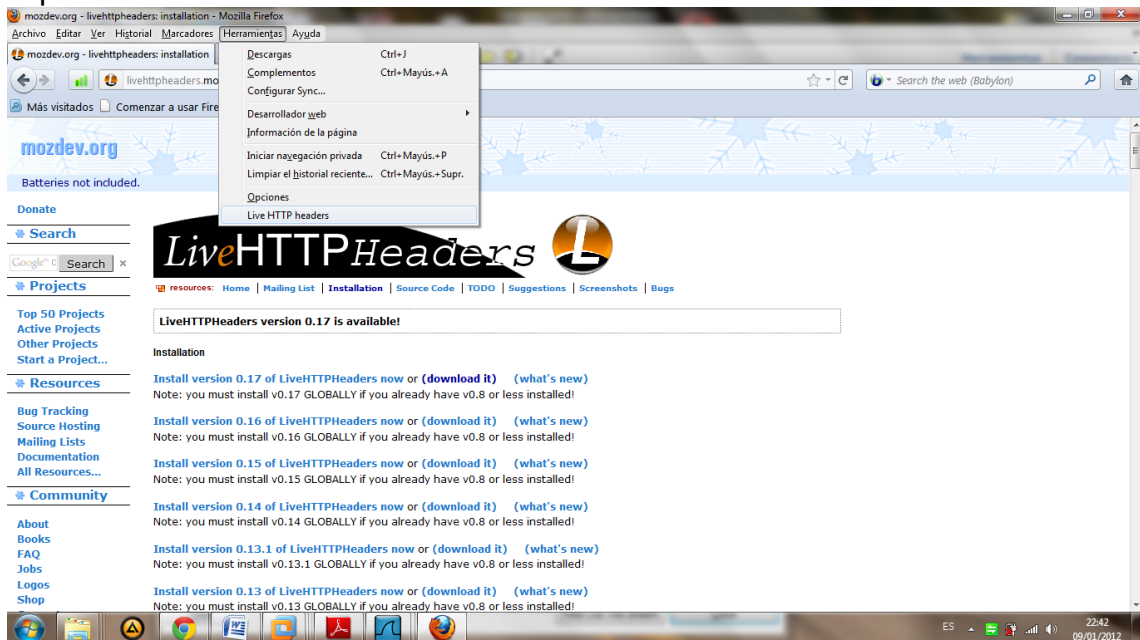
.PNG
..
IHDR...7.....4H8.....tEXtSoftware.Adobe
ImageReadyq.e<...PLTE.....IDAT..c0a..X..D...ck...*.
Y$...V.&.....L.....J
.C..+
#...L.....+m..a.....
.J]...IEND.B`.|

Entire conversation (14381 bytes)

```

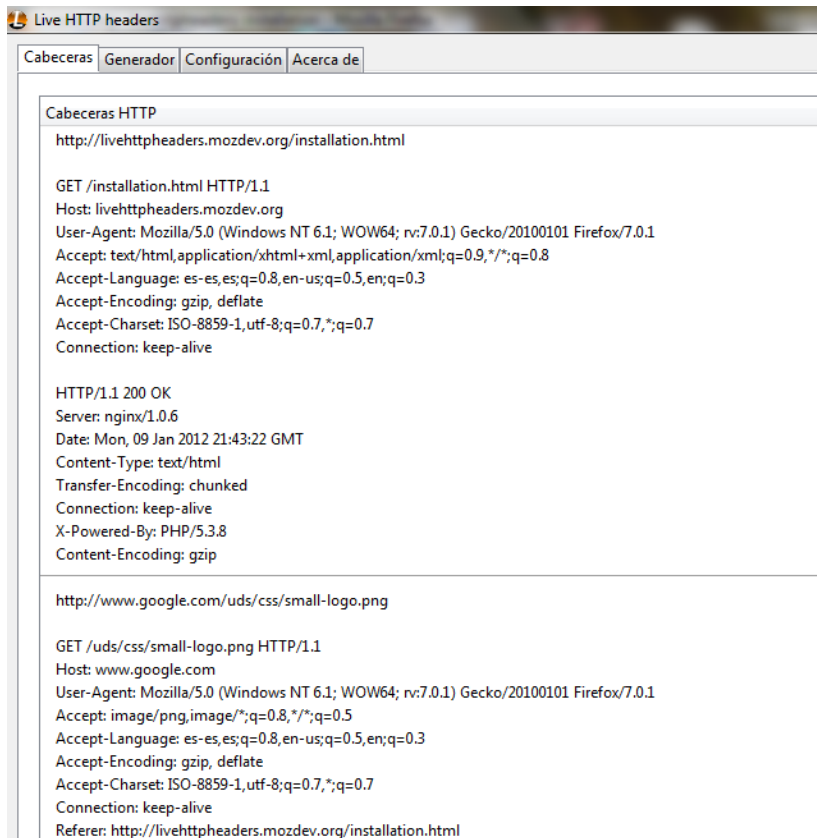
Live HTTP Header

Una vez instalado el Live HTTP Header nos dirigimos a herramientas y live http headers:



Ahora al recargar navegar por una pagina web podremos ver información sobre las misma como hemos hecho con el wireshark pero de una forma mas ordenada:

En primer lugar veremos las cabeceras donde podemos ver una gran información sobre las cabeceras de la pagina web analizada:

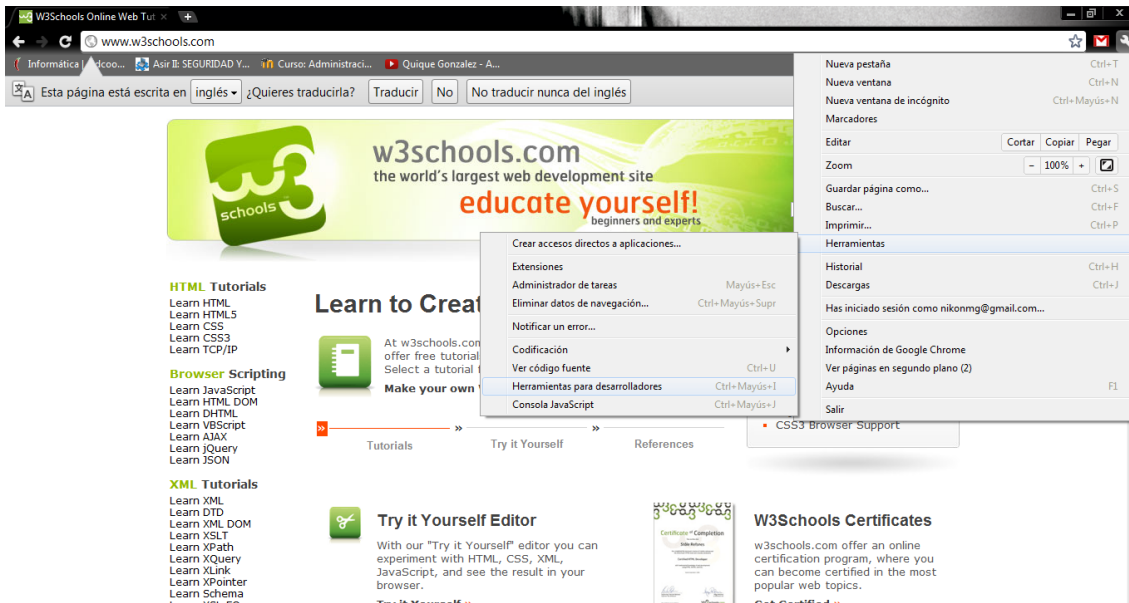


A su vez también podemos ver más información en la pestaña generador, donde podemos ver el método de obtención de los datos:

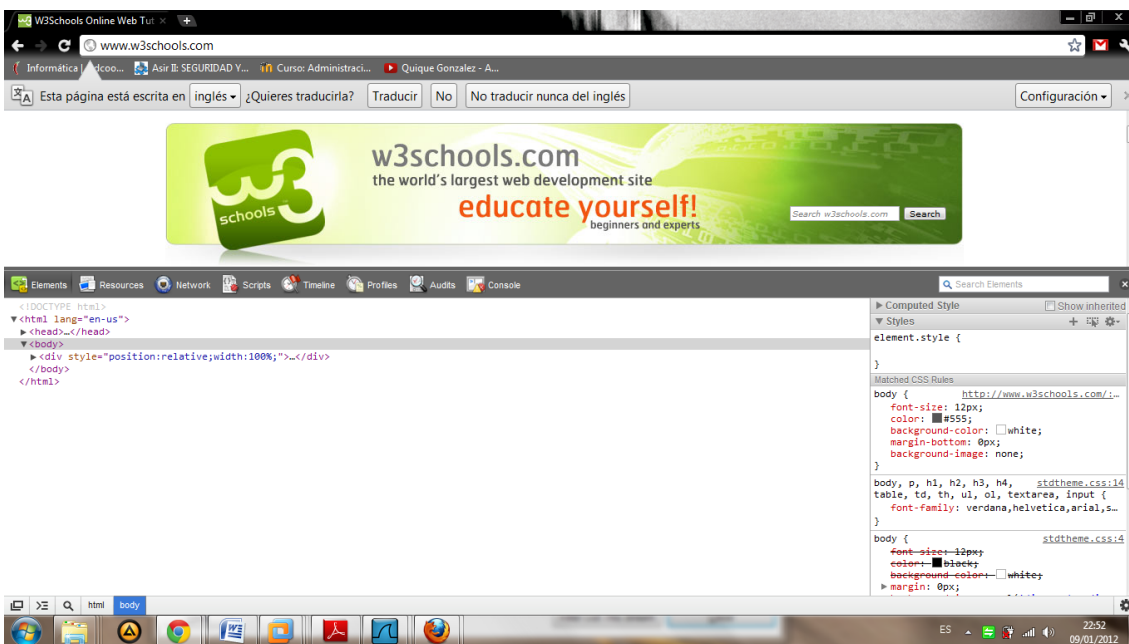


Herramientas para desarrolladores CHROME

Ahora realizaremos un análisis haciendo uso del navegador google chrome. Para ello nos dirigimos a herramientas/Herramientas para desarrolladores



Ahora en la pantalla que nos aparece podemos ver varias opciones como puede ser el código fuente de la página y el CSS de dicha página:



Actividad 3 : Instalación y configuración de un servidor HTTP (Apache2) en Ubuntu Server (configuraciones por defecto).

EN primer lugar procederemos a la instalación del apache2:

```
root@server:~# apt-get install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... 1%
```

Una vez instalado podemos comprobar como se puede observar el directorio www-data en el fichero passwd:

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
syslog:x:101:103::/home/syslog:/bin/false
primoguijarro:x:1000:1000:primoguijarro,,,:/home/primoguijarro:/bin/bash
dhcpcd:x:102:110::/var/run:/bin/false
bind:x:103:111::/var/cache/bind:/bin/false
ntp:x:104:112::/home/ntp:/bin/false
root@server:~#
```

Ahora comprobaremos que el servidor esta iniciado, para ello usamos el siguiente comando:

```
root@server:~# ps -ef | grep apache
root      1395      1   0 16:44 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1399    1395   0 16:44 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1400    1395   0 16:44 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1401    1395   0 16:44 ?        00:00:00 /usr/sbin/apache2 -k start
root      1601     832   0 16:50 tty1    00:00:00 grep --color=auto apache
root@server:~#
```

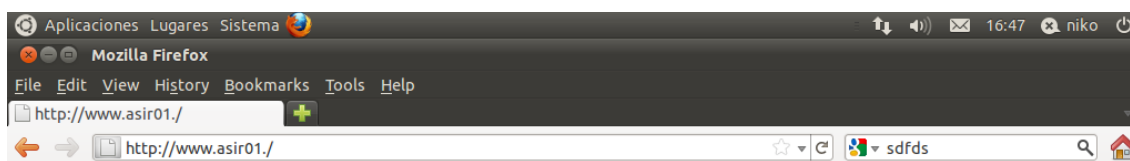
Ahora comprobaremos que el puerto 80(TCP) esta en escucha:

```
root@server:~# ps -ef | grep apache
root      1395      1   0 16:44 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1399    1395   0 16:44 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1400    1395   0 16:44 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1401    1395   0 16:44 ?        00:00:00 /usr/sbin/apache2 -k start
root      1601    832   0 16:50 tty1    00:00:00 grep --color=auto apache
root@server:~# netstat -ltn
Conexiones activas de Internet (solo servidores)
Proto Recib Enviad Dirección local      Dirección remota      Estado
tcp    0      0 0.0.0.0:80          0.0.0.0:*              ESCUCHAR
tcp    0      0 0.0.0.0:53          0.0.0.0:*              ESCUCHAR
tcp    0      0 0.0.0.0:1:53        0.0.0.0:*              ESCUCHAR
tcp    0      0 0.0.0.0:1:953       0.0.0.0:*              ESCUCHAR
tcp6   0      0 :::53               :::*                   ESCUCHAR
tcp6   0      0 :::1:953            :::*                   ESCUCHAR
root@server:~#
```

Ahora nos dirigiremos a la ruta para /var/www para ver si existe el archivo index:

```
root@server:/var/www# ls
datos index.html red.html
root@server:/var/www#
```

Para comprobar el funcionamiento del servidor usaremos un cliente que accederá al servidor web haciendo uso del alias configurado en el DNS en nuestro caso www.asir01.



It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.



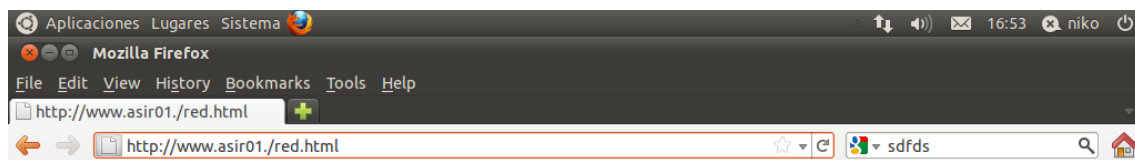
Puesto que el nombre de nuestro equipo esta definido tanto en nuestro fichero hosts como en el fichero hostname no configuraremos dicho ficheros.

Ahora procederemos a crear un archivo en el directorio /var/www con el nombre red.html con el siguiente contenido:

```
GNU nano 2.2.4 Archivo: /var/www/red.html Modificado
<html>
<body>
<h1>ESTE ES EL SERVIDOR DE LA RED ASIR01. </h1>
</body>
</html>
```

Ver ayuda Guardar Leer Fich RePág. Cortar Tex Pos actual
Salir Justificar Buscar Pág. Sig. PegarTxt Ortografía

Ahora procederemos a comprobar el funcionamiento de dicho fichero desde un cliente:



ESTE ES EL SERVIDOR DE LA RED ASIR01.

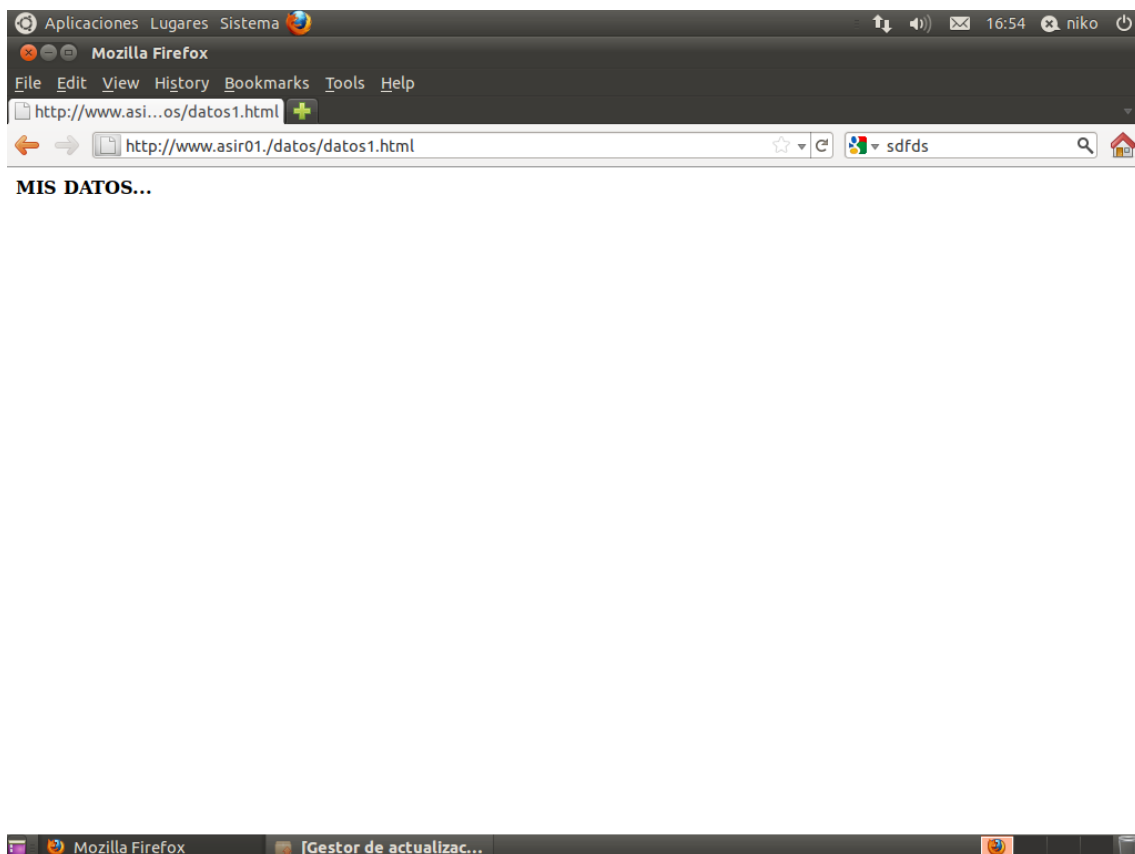
Ahora crearemos un directorio llamado datos en la ruta /var/www y que contendrá un archivo llamado datos1.html, dicho archivo contendrá lo siguiente:

```
GNU nano 2.2.4 Archivo: /var/www/datos/datos1.html
<html>
<body>
<b>MIS DATOS...</b>
</body>
</html>

[ 5 líneas escritas ]

root@server:/var/www#
```

Ahora procederemos a la comprobación del funcionamiento del archivo haciendo uso de un cliente:



El número máximo de peticiones simultaneas es de 150 y este comportamiento lo definen las directivas de nivel 1

```
GNU nano 2.2.4 Archivo: /etc/apache2/apache2.conf
# ThreadLimit: ThreadsPerChild can be changed to this maximum value during a
# graceful restart. ThreadLimit can only be changed by stopping
# and starting Apache.
# ThreadsPerChild: constant number of worker threads in each server process
# MaxRequestsPerChild: maximum number of requests a server process serves
<IfModule mpm_worker_module>
  StartServers      2
  MinSpareThreads  25
  MaxSpareThreads  75
  ThreadLimit      64
  ThreadsPerChild  25
  MaxClients       150
  MaxRequestsPerChild  0
</IfModule>

# event MPM
# StartServers: initial number of server processes to start
# MaxClients: maximum number of simultaneous client connections
# MinSpareThreads: minimum number of worker threads which are kept spare
# MaxSpareThreads: maximum number of worker threads which are kept spare

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

No se permiten conexiones persistentes, la directiva que define este comportamiento es la 3.

EN esta pantalla podemos ver el archivo donde se configura el directorio de contenido de apache:

```
GNU nano 2.2.4 Archivo: /etc/apache2/sites-available/default
<VirtualHost *:80>
  ServerAdmin webmaster@localhost

  DocumentRoot /var/www
  <Directory />
    Options FollowSymLinks
    AllowOverride None
  </Directory>
  <Directory /var/www/>
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
  </Directory>

  ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
  <Directory "/usr/lib/cgi-bin">
    AllowOverride None
    Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
    Order allow,deny
  </Directory>

[ 41 líneas leídas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

Actividad 5: Administración servidor Web HTTP (Apache2) en Ubuntu Server : Directorio por Defecto "Directory Index" -- Opciones sobre Directorios --Logs --Códigos de error.

En primer renombraremos el archivo index.html por indice.html; para ello deberemos de ejecutar el siguiente comando:

```
root@(none):~# mv /var/www/index.html /var/www/indice.html
```

Ahora desde un cliente accederemos con el navegador al servidor web, pero como el servidor no encuentra el fichero index.html muestra el contenido del directorio:



Index of /

Name	Last modified	Size	Description
datos/	09-Jan-2012 16:58	-	
indice.html	09-Jan-2012 16:44	177	
red.html	09-Jan-2012 16:56	78	

Apache/2.2.16 (Ubuntu) Server at www.asir01 Port 80

Para solucionar esta situación editaremos el archivo de configuración /etc/apache2/sites-available/default y dejamos el fichero tal y como se aprecia en la imagen, añadiendo la línea indicada:

```
GNU nano 2.2.4 Archivo: /etc/apache2/sites-available/default
<VirtualHost *:80>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/>
        DirectoryIndex red.html
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

    ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
    <Directory "/usr/lib/cgi-bin">
        AllowOverride None
        Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
    </Directory>
</VirtualHost>
```

Ahora reiniciamos el servidor para que los cambios se realicen:

```
root@none):~# service apache2 restart
```

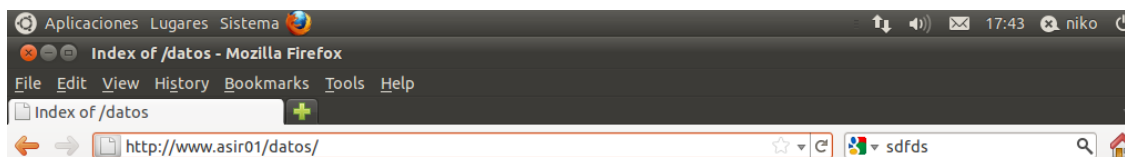
Ahora volveremos a acceder al servidor web desde un cliente y observaremos como el fichero red.html es la página de inicio:



ESTE ES EL SERVIDOR DE LA RED ASIR01.



Sin embargo si accedemos al directorio www.asir01/datos no hereda el archivo index por lo que se muestra el árbol de directorios:



Index of /datos

Name	Last modified	Size	Description
Parent Directory	-	-	-
datos1.html	09-Jan-2012 16:58	50	

Apache/2.2.16 (Ubuntu) Server at www.asir01 Port 80

Para solucionar esta situación crearemos una nueva directiva con las opciones que se muestran en pantalla:

```
GNU nano 2.2.4 Archivo: /etc/apache2/sites-available/default Modificado
    AllowOverride None
    Order allow,deny
    allow from all
</Directory>
<Directory /var/www/datos>
    DirectoryIndex index.html
    Options FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
</Directory>
ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
<Directory "/usr/lib/cgi-bin">
    AllowOverride None
    Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
    Order allow,deny
    Allow from all
</Directory>
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora reiniciamos el servidor para que los cambios se realicen:

```
root@none):~# service apache2 restart
```

Ahora desde un cliente comprobaremos el funcionamiento de los cambios efectuados:



LOGS DE ERROR

En primer lugar consultaremos el archivo `/etc/apache2/sites-available/default`.

Una vez allí podemos observar donde están los log de errores y el log de accesos:

```
GNU nano 2.2.4 Archivo: /etc/apache2/sites-available/default

</Directory>

ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
<Directory "/usr/lib/cgi-bin">
    AllowOverride None
    Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
    Order allow,deny
    Allow from all
</Directory>

ErrorLog ${APACHE_LOG_DIR}/error.log

# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
LogLevel warn

CustomLog ${APACHE_LOG_DIR}/access.log combined

Alias /doc/ "/usr/share/doc/"
<Directory "/usr/share/doc/">
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora consultaremos el log de errores para ello deberemos de ejecutar el siguiente comando:

```
Cat /var/log/apache2/error.log
```

Una vez allí podemos observar los diversos errores que se han producido:

```
g normal operations
[Mon Jan 09 17:21:08 2012] [notice] caught SIGTERM, shutting down
[Mon Jan 09 17:21:09 2012] [notice] Apache/2.2.16 (Ubuntu) configured -- resumin
g normal operations
[Mon Jan 09 17:21:48 2012] [notice] caught SIGTERM, shutting down
[Mon Jan 09 17:21:49 2012] [notice] Apache/2.2.16 (Ubuntu) configured -- resumin
g normal operations
[Mon Jan 09 17:10:21 2012] [notice] Apache/2.2.16 (Ubuntu) configured -- resumin
g normal operations
[Mon Jan 09 17:10:43 2012] [notice] caught SIGTERM, shutting down
[Mon Jan 09 17:10:44 2012] [notice] Apache/2.2.16 (Ubuntu) configured -- resumin
g normal operations
[Mon Jan 09 17:28:30 2012] [notice] caught SIGTERM, shutting down
[Mon Jan 09 17:28:31 2012] [notice] Apache/2.2.16 (Ubuntu) configured -- resumin
g normal operations
[Mon Jan 09 17:31:22 2012] [error] [client 10.33.1.5] File does not exist: /var/
www/favicon.ico
[Mon Jan 09 17:31:22 2012] [error] [client 10.33.1.5] File does not exist: /var/
www/favicon.ico
[Mon Jan 09 17:37:50 2012] [notice] caught SIGTERM, shutting down
[Mon Jan 09 17:37:51 2012] [notice] Apache/2.2.16 (Ubuntu) configured -- resumin
g normal operations
[Mon Jan 09 17:37:57 2012] [error] [client 10.33.1.5] Directory index forbidden
by Options directive: /var/www/datos/
root@(none):~# _
```


Ahora consultaremos el log de accesos para ello deberemos de ejecutar el siguiente comando:

```
Cat /var/log/apache2/access.log
```

Una vez allí podemos observar información sobre los clientes como es el navegador que ha utilizado para acceder al servidor web:

```
10.33.1.5 - - [09/Jan/2012:17:22:57 +0100] "GET /icons/text.gif HTTP/1.1" 200 520 "http://www.asir01./" "Mozilla/5.0 (X11; Linux i686; rv:2.0) Gecko/20100101 Firefox/4.0"
10.33.1.5 - - [09/Jan/2012:17:29:59 +0100] "GET / HTTP/1.1" 200 423 "-" "Mozilla/5.0 (X11; Linux i686; rv:2.0) Gecko/20100101 Firefox/4.0"
10.33.1.5 - - [09/Jan/2012:17:31:22 +0100] "GET /datos HTTP/1.1" 301 552 "-" "Mozilla/5.0 (X11; Linux i686; rv:2.0) Gecko/20100101 Firefox/4.0"
10.33.1.5 - - [09/Jan/2012:17:31:22 +0100] "GET /datos/ HTTP/1.1" 200 699 "-" "Mozilla/5.0 (X11; Linux i686; rv:2.0) Gecko/20100101 Firefox/4.0"
10.33.1.5 - - [09/Jan/2012:17:31:22 +0100] "GET /icons/blank.gif HTTP/1.1" 200 438 "http://www.asir01/datos/" "Mozilla/5.0 (X11; Linux i686; rv:2.0) Gecko/20100101 Firefox/4.0"
10.33.1.5 - - [09/Jan/2012:17:31:22 +0100] "GET /icons/text.gif HTTP/1.1" 200 519 "http://www.asir01/datos/" "Mozilla/5.0 (X11; Linux i686; rv:2.0) Gecko/20100101 Firefox/4.0"
10.33.1.5 - - [09/Jan/2012:17:31:22 +0100] "GET /icons/back.gif HTTP/1.1" 200 507 "http://www.asir01/datos/" "Mozilla/5.0 (X11; Linux i686; rv:2.0) Gecko/20100101 Firefox/4.0"
10.33.1.5 - - [09/Jan/2012:17:31:22 +0100] "GET /favicon.ico HTTP/1.1" 404 501 "-" "Mozilla/5.0 (X11; Linux i686; rv:2.0) Gecko/20100101 Firefox/4.0"
10.33.1.5 - - [09/Jan/2012:17:31:22 +0100] "GET /favicon.ico HTTP/1.1" 404 501 "-" "Mozilla/5.0 (X11; Linux i686; rv:2.0) Gecko/20100101 Firefox/4.0"
10.33.1.5 - - [09/Jan/2012:17:37:57 +0100] "GET /datos/ HTTP/1.1" 403 502 "-" "Mozilla/5.0 (X11; Linux i686; rv:2.0) Gecko/20100101 Firefox/4.0"
root@(none):~#
```

Código del error(ErrorDocument)

Ahora configuraremos un servidor virtual por defecto para que retorne el código de error 404 y envíe el texto "Página no encontrada en el servidor asir01.". Para ello editaremos el archivo /etc/apache2/sites/available/default como se muestra en la imagen:

```
GNU nano 2.2.4 Archivo: /etc/apache2/sites-available/default

</Directory>

ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
<Directory "/usr/lib/cgi-bin">
    AllowOverride None
    Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
    Order allow,deny
    Allow from all
</Directory>

ErrorLog /var/log/apache2/error.log

# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
LogLevel warn

CustomLog /var/log/apache2/access.log combined

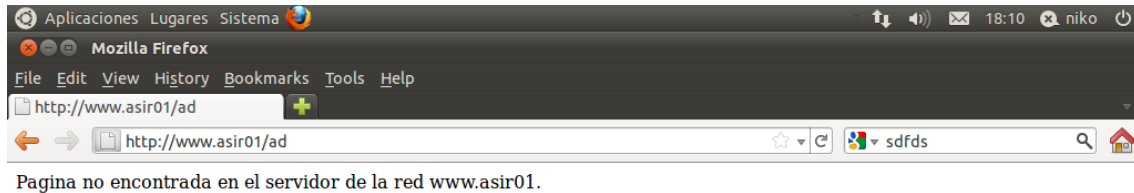
ErrorDocument_404 "Página no encontrada en el servidor de la red www.asir01."

[ 52 líneas escritas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^I Ortografía
```

Ahora reiniciamos el servidor para que los cambios se realicen:

```
root@none):~# service apache2 restart_
```

Ahora desde un cliente intentaremos acceder a una pagina que no exista en nuestro servidor y nos debe de aparecer el siguiente mensaje:



Ahora crearemos la pagina `noencontrada.html` con el contenido "La pagina solicitada no ha sido encontrada" con el objetivo que cada vez que no se encuentre una pagina y de el error 404 sea redirigido a ese archivo. Para ello en primer lugar crearemos el archivo `noencontrada.html`:

```
GNU nano 2.2.4 Archivo: /var/www/noencontrada.html
<html>
<body>
<p>La pagina solicitada no ha sido encontrada</p>_
</body>
</html>

[ 6 líneas escritas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora configuraremos el archivo de configuración de apache2 con el objetivo de que se redireccione a ese archivo cada vez que ocurra ese error:

```
GNU nano 2.2.4 Archivo: /etc/apache2/sites-available/default

# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
LogLevel warn

CustomLog /var/log/apache2/access.log combined
ErrorDocument 404 /noencontrada.html_

Alias /doc/ "/usr/share/doc/"
<Directory "/usr/share/doc/">
  Options Indexes MultiViews FollowSymLinks
  AllowOverride None
  Order deny,allow
  Deny from all
  Allow from 127.0.0.0/255.0.0.0 ::1/128
</Directory>
</VirtualHost>

[ 52 líneas escritas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora reiniciamos el servidor para que los cambios se realicen:

```
root@(none):~# service apache2 restart
```

Ahora desde un cliente intentaremos acceder a una pagina que no exista en nuestro servidor y nos debe de aparecer el siguiente mensaje:



Actividad 6: Administrador servidor Web HTTP (Apache2) en Ubuntu Server: -- Directorios Virtuales (*Alias y enlaces simbólicos*).

En primer lugar en nuestro directorio de conexión nos crearemos 2 directorios cuyos nombres serán wiki y blog:

```
root@(none):/home/niko# mkdir wiki blog
root@(none):/home/niko# ls
blog  wiki
```

Ahora en el directorio wiki creamos un archivo llamado wiki1.html cuyo contenido será "Página wiki 1":

```
GNU nano 2.2.4 Archivo: /home/niko/wiki/wiki1.html
<html>
<body>
<p>PAGINA WIKI 1</p>_
</body>
</html>
```

[5 líneas escritas]

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía

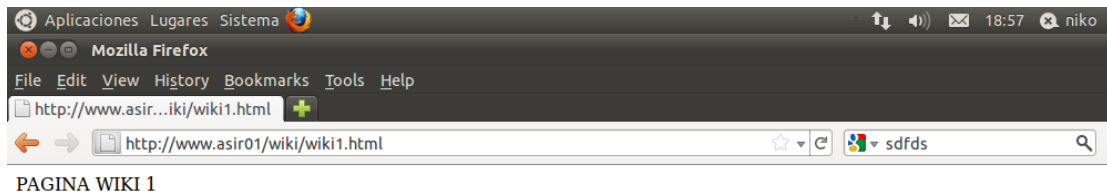
Ahora nos dirigimos al archivo `/etc/apache2/sites-available/default` y allí crearemos un directorio denominado wiki que haga referencia a la ruta `/home/niko/wiki`:

```
Alias /wiki /home/niko/wiki
<Directory /home/niko/wiki>
    DirectoryIndex index.html
    Options FollowSymlinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
</Directory>
```

Ahora reiniciamos el servidor para que los cambios se realicen:

```
root@(none):~# service apache2 restart
```

Ahora desde un cliente intentaremos acceder a una pagina que no exista en nuestro servidor y nos debe de aparecer el siguiente mensaje que es el que se encuentra en el archivo wiki1.html:



Ahora en el directorio /home/blog creamos un archivo llamado blog1.html que contendrá el texto "PAGINA BLOG 1":

```
GNU nano 2.2.4 Archivo: /home/niko/blog/blog1.html
<html>
<body>
<p1> PAGINA BLOG 1</p1>_
</body>
</html>
[ 5 líneas escritas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora crearemos un enlace simbólico /home/niko/blog que apunte a /var/www/blog:

```
oot@(none):/home/niko# ln -s /home/niko/blog /var/www/blog
```

Ahora nos dirigimos al archivo `/etc/apache2/sites-available/default` realizamos las siguientes configuraciones:

```
GNU nano 2.2.4 Archivo: /etc/apache2/sites-available/default

</Directory>

Alias /wiki /home/niko/wiki
<Directory /home/niko/wiki>
    DirectoryIndex index.html
    Options FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
</Directory>

<Directory /var/www/blog>
    DirectoryIndex index.html
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
</Directory>_

ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
[ 69 líneas escritas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^T PegarTxt ^T Ortografía
```

Ahora reiniciamos el servidor para que los cambios se realicen:

```
root@none):~# service apache2 restart
```

Ahora desde un cliente intentaremos acceder a una pagina que no exista en nuestro servidor y nos debe de aparecer el siguiente mensaje que es el que se encuentra en el directorio blog:



```
Aplicaciones Lugares Sistema
Index of /blog - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Index of /blog
http://www.asir01/blog/
Index of /blog
Name Last modified Size Description
Parent Directory -
blog1.html 09-Jan-2012 18:42 55
Apache/2.2.16 (Ubuntu) Server at www.asir01 Port 80
```

Actividad 7: Administración servidor Web HTTP (Apache2) en Ubuntu Server: --Módulos -- Directorios personales de usuarios .

En primer lugar comprobaremos los módulos estaticos que se han cargado al compilar el servidor, para ello ejecutamos el siguiente comando `apache2ctl -l`:

```
root@(none):/home/niko# apache2ctl -l
Compiled in modules:
  core.c
  mod_log_config.c
  mod_logio.c
  worker.c
  http_core.c
  mod_so.c
```

Ahora comprobaremos los módulos que se han cargado dinámicamente al arrancar el servidor consultando el directorio `/etc/apache2/mods-enabled`

```
root@(none):/etc/apache2/mods-enabled# ls
alias.conf          authz_user.load    dir.conf           reqtimeout.conf
alias.load          autoindex.conf    dir.load           reqtimeout.load
auth_basic.load     autoindex.load    env.load           setenvif.conf
authn_file.load     cgid.conf         mime.conf          setenvif.load
authz_default.load  cgid.load         mime.load          status.conf
authz_groupfile.load deflate.conf       negotiation.conf  status.load
authz_host.load     deflate.load       negotiation.load
```

Ahora editaremos el fichero `alias.load`, dentro de este archivo podemos observar la ruta del código del módulo `.so`

```
GNU nano 2.2.4          Archivo: alias.load
LoadModule alias_module /usr/lib/apache2/modules/mod_alias.so
```

Ahora editaremos un archivo `.conf` en mi caso será el `alias.conf`, allí podemos observar la declaración `<IfModule>` que se ejecutara si se carga el módulo:

```
# We include the /icons/ alias for FancyIndexed directory listings.  If
# you do not use FancyIndexing, you may comment this out.
#
Alias /icons/ "/usr/share/apache2/icons/"

<Directory "/usr/share/apache2/icons">
  Options Indexes MultiViews
  AllowOverride None
  Order allow,deny
  Allow from all
</Directory>

</IfModule>
```


Ahora consultaremos el directorio `/usr/lib/apache2/modules` para observar los módulos a cargar:

```
root@(none):/usr/lib/apache2/modules# ls
httpd.exp                mod_cgi.so              mod_mime_magic.so
mod_actions.so          mod_charset_lite.so    mod_mime.so
mod_alias.so            mod_dav_fs.so          mod_negotiation.so
mod_asis.so             mod_dav_lock.so        mod_proxy_ajp.so
mod_auth_basic.so       mod_dav.so             mod_proxy_balancer.so
mod_auth_digest.so      mod_dbd.so             mod_proxy_connect.so
mod_authn_alias.so      mod_deflate.so         mod_proxy_ftp.so
mod_authn_anon.so       mod_dir.so             mod_proxy_http.so
mod_authn_dbd.so        mod_disk_cache.so      mod_proxy_scgi.so
mod_authn_dbm.so        mod_dumpio.so          mod_proxy.so
mod_authn_default.so    mod_env.so             mod_reqtimeout.so
mod_authn_file.so       mod_expires.so         mod_rewrite.so
mod_authnz_ldap.so      mod_ext_filter.so      mod_setenvif.so
mod_authz_dbm.so        mod_file_cache.so      mod_spelling.so
mod_authz_default.so    mod_filter.so          mod_ssl.so
mod_authz_groupfile.so  mod_headers.so         mod_status.so
mod_authz_host.so       mod_ident.so           mod_substitute.so
mod_authz_owner.so      mod_imagemap.so        mod_suexec.so
mod_authz_user.so       mod_include.so         mod_unique_id.so
mod_autoindex.so        mod_info.so            mod_userdir.so
mod_cache.so            mod_ldap.so            mod_usertrack.so
mod_cern_meta.so        mod_log_forensic.so    mod_version.so
mod_cgid.so             mod_mem_cache.so       mod_vhost_alias.so
root@(none):/usr/lib/apache2/modules# _
```

También podemos ejecutar `apt-cache search libapache2-mod`, para ver los módulos adicionales de apache que se pueden instalar en nuestra distribución de Linux:

```
libapache2-mod-log-sql - Use SQL to store/write your apache queries logs - Base
libapache2-mod-log-sql-dbi - Use SQL to store/write your apache queries logs - D
BI interface
libapache2-mod-log-sql-mysql - Use SQL to store/write your apache queries logs -
MySQL interface
libapache2-mod-mono - Apache module for running ASP.NET applications on Mono
libapache2-mod-musicindex - Browse, stream, download and search through MP3/Ogg/
FLAC files
libapache2-mod-neko - Apache module for running server-side Neko programs
libapache2-mod-removeip - Module to remove IP from apache2's logs
libapache2-mod-suphp - Módulo Apache2 para correr guiones php con permisos de pr
opietario
mod-musicindex-common - Archivos comunes para mod-musicindex
libapache2-mod-php5 - server-side, HTML-embedded scripting language (Apache 2 mo
dule)
php5-cgi - server-side, HTML-embedded scripting language (CGI binary)
libapache2-mod-apparmor - changehat AppArmor library as an Apache module
libapache2-mod-authnz-external - authenticate Apache against external authentica
tion services
libapache2-mod-fcgid - an alternative module compat with mod_fastcgi
libapache2-mod-fcgid-dbg - debugging symbols for mod_fcgid
libapache2-mod-php5filter - server-side, HTML-embedded scripting language (apach
e 2 filter module)
php5-fpm - server-side, HTML-embedded scripting language (FPM-CGI binary)
root@(none):/usr/lib/apache2/modules# _
```

Directorios personales de usuarios (modulo userdir)

En primer lugar comprobaremos si el modulo userdir esta activado, para ello nos dirigimos al directorio /etc/apache2/mods-enabled y comprobamos que esta dicho modulo:

```
root@(none):/usr/lib/apache2/modules# cd /etc/apache2/mods-enabled
root@(none):/etc/apache2/mods-enabled# ls
alias.conf          authz_user.load    dir.conf           reqtimeout.conf
alias.load          autoindex.conf    dir.load           reqtimeout.load
auth_basic.load     autoindex.load    env.load           setenvif.conf
authn_file.load     cgid.conf         mime.conf         setenvif.load
authz_default.load  cgid.load         mime.load         status.conf
authz_groupfile.load deflate.conf      negotiation.conf  status.load
authz_host.load     deflate.load      negotiation.load
root@(none):/etc/apache2/mods-enabled#
```

Como no está activado procederemos a activarlo, para ello ejecutamos el comando de la siguiente imagen:

```
root@(none):/etc/apache2/mods-enabled# a2enmod userdir
Enabling module userdir.
Run '/etc/init.d/apache2 restart' to activate new configuration!
root@(none):/etc/apache2/mods-enabled#
```

Ahora deberemos de reiniciar el servidor para que los cambios se efectúen:

```
root@(none):~# service apache2 restart
```

Ahora podemos observar que esta en el directorio /etc/apache2/mods-enabled:

```
root@(none):/etc/apache2/mods-enabled# ls
alias.conf          autoindex.conf    env.load           setenvif.load
alias.load          autoindex.load    mime.conf         status.conf
auth_basic.load     cgid.conf         mime.load         status.load
authn_file.load     cgid.load         negotiation.conf  userdir.conf
authz_default.load  deflate.conf      negotiation.load  userdir.load
authz_groupfile.load deflate.load      reqtimeout.conf
authz_host.load     dir.conf          reqtimeout.load
authz_user.load     dir.load         setenvif.conf
root@(none):/etc/apache2/mods-enabled#
```

Ahora editamos el archivo userdir.conf:

```
GNU nano 2.2.4 Archivo: /etc/apache2/mods-enabled/userdir.conf
<IfModule mod_userdir.c>
  UserDir public_html
  UserDir disabled root

  <Directory /home/*/public_html>
    AllowOverride FileInfo AuthConfig Limit Indexes
    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
    <Limit GET POST OPTIONS>
      Order allow,deny
      Allow from all
    </Limit>
    <LimitExcept GET POST OPTIONS>
      Order deny,allow
      Deny from all
    </LimitExcept>
  </Directory>
</IfModule>

[ 18 líneas leídas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora crearemos el directorio public_html en /home/niko; dentro de ese directorio crearemos un archivo llamado perosnal1.html y tendrá como contenido el texto “PAGINA PERSONAL DEL USUARIO ALUMNO”.

```
niko@(none):/etc/apache2/mods-enabled$ mkdir /home/niko/public_html
niko@(none):/etc/apache2/mods-enabled$
```

Ahora crearemos el archivo personal1.html:

```
GNU nano 2.2.4 Archivo: /home/niko/public_html/personal1.html
<html>
<body>
<p1>PAGINA PERSONAL DEL USUARIO ALUMNO</p1>
</body>
</html>

[ 5 líneas escritas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora reiniciamos el servidor para que los cambios se realicen:

```
root@(none):~# service apache2 restart
```

Ahora desde un cliente intentaremos acceder a una pagina que no exista en nuestro servidor y nos debe de aparecer el siguiente mensaje que es el que se encuentra en el directorio public_html, para ello he introducido la línea `www.asir01./~niko/personal1.html`:



PAGINA PERSONAL DEL USUARIO ALUMNO

Actividad 8: Administración servidor Web HTTP (Apache2) en Ubuntu Server: --Control de acceso por IP -- Autenticación HTTP Basic.

En primer lugar crearemos el directorio `/var/www/privado`, dentro de este directorio crearemos un fichero cuyo contenido será el siguiente texto "PAGINA PRIVADA"

```
GNU nano 2.2.4 Archivo: /var/www/privado/privado1.html Modificado
<html>
<body>
<p>PAGINA PRIVADA </p>_
</body>
</html>

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

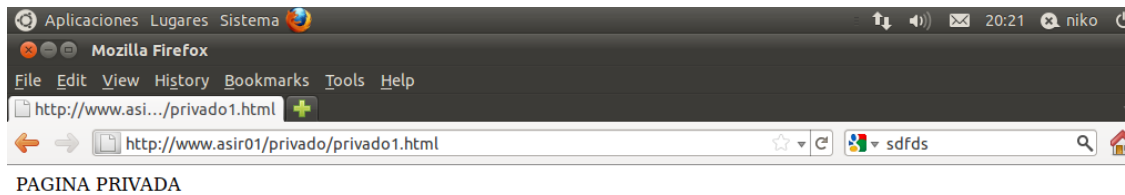
Despues editaremos el fichero de configuración `/etc/apache2/sites-available/default` y introducimos la siguiente configuración para denegar el acceso a todos los usuarios excepto al equipo local y al cliente Ubuntu:

```
<Directory /var/www/privado>_
Options Indexes FollowSymLinks MultiViews
AllowOverride None
Order allow,deny
allow from 127.0.0.1 localhost
allow from 10.33.1.5
</Directory>
```

Ahora reiniciamos el servidor para que los cambios se realicen:

```
root@none):~# service apache2 restart
```

Ahora desde un cliente intentaremos acceder a una pagina que no exista en nuestro servidor y nos debe de aparecer el siguiente mensaje:



AUTENTICACIÓN HTTP BASIC

Ahora deberemos de dirigirnos al directorio `/etc/apache2/mods-enabled` para comprobar si el modulo `auth_basic` esta activado:

```
root@(none):/etc/apache2/mods-enabled# ls
alias.conf          autoindex.conf    env.load          setenvif.load
alias.load          autoindex.load    mime.conf         status.conf
auth_basic.load     cgid.conf         mime.load         status.load
authn_file.load     cgid.load         negotiation.conf  userdir.conf
authz_default.load  deflate.conf      negotiation.load  userdir.load
authz_groupfile.load deflate.load       reqtimeout.conf
authz_host.load     dir.conf          reqtimeout.load
authz_user.load     dir.load          setenvif.conf
root@(none):/etc/apache2/mods-enabled#
```

Ahora crearemos un fichero que sea accesible por apache en el que guardaremos los usuarios y sus contraseñas; para ello realizamos los siguientes pasos:

En primer lugar ejecutamos el comando `htpasswd -c /etc/apache2/passwd niko` (esto es el usuario)

```
root@(none):/etc/apache2/mods-enabled# htpasswd -c /etc/apache2/passwd niko
New password:
Re-type new password:
Adding password for user niko
root@(none):/etc/apache2/mods-enabled#
```

Ahora editaremos el fichero de configuración `/etc/apache2/sites-available/default` y permitimos el acceso al usuario niko:

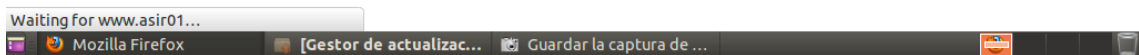
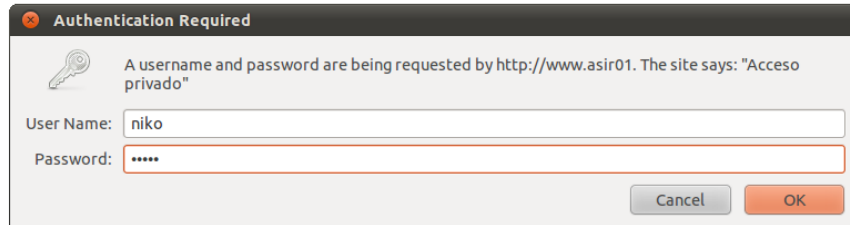
```
<Directory /var/www/privado>
  Options Indexes FollowSymLinks MultiViews
  AllowOverride None
  Order allow,deny
  allow from 127.0.0.1 localhost
  allow from 10.33.1.5
  AuthName "Acceso privado"
  AuthType Basic
  AuthUserFile /etc/apache2/passwd
  Require user niko_
</Directory>
```

Ahora
reinici
amos

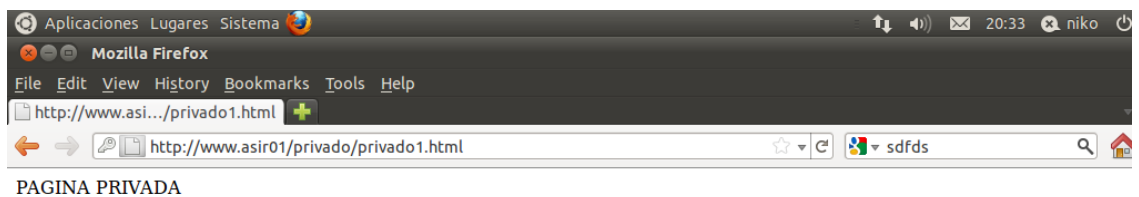
```
root@(none):~# service apache2 restart_
```

el servidor para que los cambios se realicen:

Ahora desde un cliente intentaremos acceder a una pagina para la cual nos debe de salir un formulario en el que introducir el usuario y la contraseña:



Una vez introducido el usuario y la contraseña podremos visualizar la web:



Actividad 9: Administración servidor Web HTTP (Apache2) en Ubuntu Server: --Fichero .htaccess

En primer lugar editaremos el fichero de configuración `/etc/apache2/sites-available/default` y introducimos la siguiente configuración para habilitar el uso de ficheros `.htaccess` permitiendo así sobrescribir todas las directivas en el directorio `/home/niko/wiki`.

```
Alias /wiki /home/niko/wiki
<Directory /home/niko/wiki>
    AllowOverride All
</Directory>
```

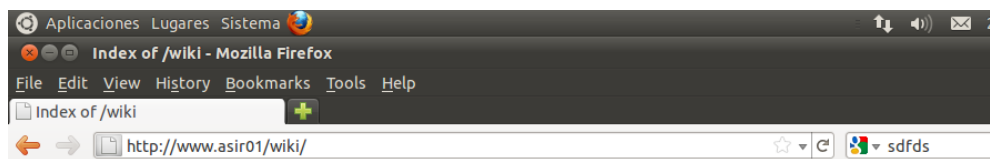
Ahora creamos el fichero `.htaccess` en la ruta `/home/niko/wiki`. Una vez creado dicho archivo debe de contener lo siguiente para poder permitir el acceso al equipo local ya todos los equipos de la red:

```
GNU nano 2.2.4 Archivo: /home/niko/wiki/.htaccess
DirectoryIndex Index.html
Options Indexes FollowSymlinks MultiViews
Order allow,deny
allow from localhost 127.0.0.1
allow from 10.33.1.0/24_
```

Ahora reiniciamos el servidor para que los cambios se realicen e intentamos acceder al recurso `wiki` desde un cliente:

```
root@none):~# service apache2 restart_
```

Como se aprecia en la imagen podemos acceder:



Index of /wiki

Name	Last modified	Size	Description
Parent Directory		-	
wiki1.html	09-Jan-2012 18:23	53	

Apache/2.2.16 (Ubuntu) Server at www.asir01 Port 80

Ahora creamos el archivo `.htpasswd` en el directorio `/home/niko/.htpasswd` y en dicho archivo añadiremos al usuario `wiki`:

```
root@(none):/etc/apache2/mods-enabled# htpasswd -c /home/niko/.htpasswd wiki
New password:
Re-type new password:
Adding password for user wiki
root@(none):/etc/apache2/mods-enabled#
```

Ahora editaremos el fichero `/home/niko/wiki/.htaccess` y añadimos las siguientes líneas:

```
GNU nano 2.2.4 Archivo: /home/niko/wiki/.htaccess
DirectoryIndex Index.html
Options Indexes FollowSymlinks MultiViews
Order allow,deny
allow from localhost 127.0.0.1
allow from 10.33.1.0/24
AuthName "Acceso privado a la wiki"
AuthType Basic
AuthUserFile /home/niko/wiki/.htpasswd
Require user wiki_

[ 9 líneas escritas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora reiniciamos el servidor para que los cambios se realicen e intentamos acceder al recurso `wiki` desde un cliente:

```
root@(none):~# service apache2 restart
```

Ahora desde un cliente intentaremos acceder a una pagina para la cual nos debe de salir un formulario en el que introducir el usuario y la contraseña:



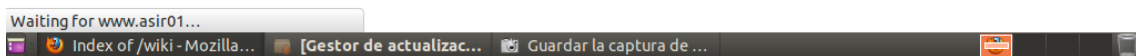
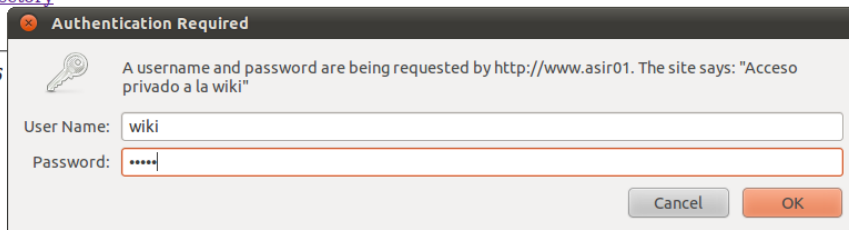
Index of /wiki

[Name](#) [Last modified](#) [Size](#) [Description](#)

[Parent Directory](#)

[wiki1.html](#)

Apache/2.2.16



Una vez logueado con el usuario wiki podremos ver el contenido del directorio:



Index of /wiki

[Name](#) [Last modified](#) [Size](#) [Description](#)

[Parent Directory](#)

[wiki1.html](#) 09-Jan-2012 18:23 53

Apache/2.2.16 (Ubuntu) Server at www.asir01 Port 80

Actividad 10: Administración servidor Web HTTP (Apache2) en Ubuntu Server: --Monitorización y logs -- Pruebas de rendimiento.

En primer lugar usaremos el modulo `mod_status`, por lo tanto deberemos de comprobar que el si el modulo activado:

```
niko@(none):/etc/apache2/mods-enabled$ ls
alias.conf          autoindex.conf    env.load          setenvif.load
alias.load          autoindex.load    mime.conf         status.conf
auth_basic.load     cgid.conf         mime.load         status.load
authn_file.load     cgid.load         negotiation.conf userdir.conf
authz_default.load  deflate.conf      negotiation.load  userdir.load
authz_groupfile.load deflate.load       reqtimeout.conf
authz_host.load     dir.conf          reqtimeout.load
authz_user.load     dir.load          setenvif.conf
niko@(none):/etc/apache2/mods-enabled$
```

Este modulos nos permite registrar información sobre el acceso y el estado del servidor web.

Para ello deberemos de configurar el archivo `status.conf` para asi poder acceder via web a log que genera este módulo; para ello deberemos de añadir la dirección del host como se muestra en las siguiente imagen:

```
GNU nano 2.2.4 Archivo: status.conf

<IfModule mod_status.c>
#
# Allow server status reports generated by mod_status,
# with the URL of http://servername/server-status
# Uncomment and change the "192.0.2.0/24" to allow access from other hosts.
#
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
    Allow from 127.0.0.1 ::1
    Allow from 10.33.1.5_
</Location>

# Keep track of extended status information for each request
ExtendedStatus On

# Determine if mod_status displays the first 63 characters of a request or
# the last 63, assuming the request itself is greater than 63 chars.
# Default: Off

[ 30 líneas escritas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora desde un cliente accederemos via web poniendo en el navegador la dirección www.asir01./server-status

En primer lugar deberemos de comprobar que el modulo mod_info esta activado:

```
niko@(none):/etc/apache2/mods-enabled$ ls
alias.conf          autoindex.conf    env.load          setenvif.load
alias.load         autoindex.load    mime.conf         status.conf
auth_basic.load    cgid.conf         mime.load        status.load
authn_file.load    cgid.load        negotiation.conf userdir.conf
authz_default.load deflate.conf      negotiation.load userdir.load
authz_groupfile.load deflate.load      reqtimeout.conf
authz_host.load    dir.conf         reqtimeout.load
authz_user.load    dir.load         setenvif.conf
niko@(none):/etc/apache2/mods-enabled$ _
```

Como el modulo no está activado deberemos de activarlo, para ello seguimos los siguientes pasos:

```
root@(none):/home/niko# a2enmod info
Enabling module info.
Run '/etc/init.d/apache2 restart' to activate new configuration!
```

Ahora editaremos el fichero info.conf y permitimos la dirección de red de los clientes:

```
<IfModule mod_info.c>
#
# Allow remote server configuration reports, with the URL of
# http://servername/server-info (requires that mod_info.c be l
# Uncomment and change the "192.0.2.0/24" to allow access from
#
<Location /server-info>
    SetHandler server-info
    Order deny,allow
    Deny from all
    Allow from 127.0.0.1 ::1
    Allow from 10.33.1_0/24
</Location>
</IfModule>
```

Ahora accederemos con un cliente y podremos ver una gran información sobre el servidor

Server Settings

Server Version: Apache/2.2.16 (Ubuntu)
Server Built: Nov 3 2011 02:38:15
Server loaded APR Version: 1.4.2
Compiled with APR Version: 1.4.2
Server loaded APU Version: 1.3.9
Compiled with APU Version: 1.3.9
Module Magic Number: 20051115:24
Hostname/port: www.asir01:80
Timeouts: connection: 300 keep-alive: 15
MPM Name: Worker
MPM Information: Max Daemons: 1 Threaded: yes Forked: yes
Server Architecture: 32-bit
Server Root: /etc/apache2
Config File: /etc/apache2/apache2.conf
Server Built With:
 -D APACHE_MPM_DIR="server/mpm/worker"
 -D APR_HAS_SENDFILE
 -D APR_HAS_MMAP
 -D APR_HAVE_IPV6 (IPv4-mapped addresses enabled)
 -D APR_USE_SYSVSEM_SERIALIZE

A su vez podremos ver la información de los módulos:

Content handlers: none
Configuration Phase Participation: none
Request Phase Participation: Pre-Connection, Logging
Module Directives: none

Module Name: [mod_log_config.c](#)
Content handlers: none
Configuration Phase Participation: Create Server Config, Merge Server Configs
Request Phase Participation: Logging
Module Directives:
 CustomLog - a file name, a custom log format string or format name, and an optional "env=" clause (see docs)
 TransferLog - the filename of the access log
 LogFormat - a log format string (see docs) and an optional format name
 CookieLog - the filename of the cookie log
 BufferedLogs - Enable Buffered Logging (experimental)

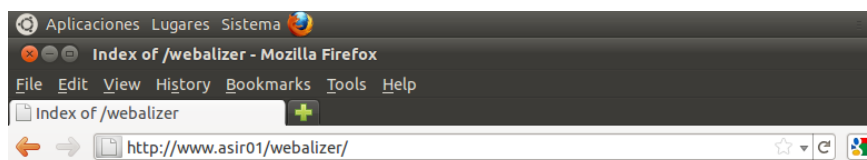
Current Configuration:
In file: /etc/apache2/apache2.conf
 217: LogFormat "%v:%p %h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" vhost_combined
 218: LogFormat "%h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" combined
 219: LogFormat "%h %l %u %t \"%r\" %>s %O" common
 220: LogFormat "%{Referer}i -> %U" referer
 221: LogFormat "%{User-agent}i" agent
In file: /etc/apache2/conf.d/other-vhosts-access-log

Herramientas gráficas de monitorización:

Webalizer

```
root@(none):~# webalizer /var/log/apache2/access.log
Webalizer V2.01-10 (Linux 2.6.35-22-generic-pae) locale: es_ES.UTF-8
Utilizando histórico /var/log/apache2/access.log (clf)
Using default GeoIP database
Creando informe en /var/www/webalizer
El nombre de máquina en el informe es '(none)'
Leyendo archivo... webalizer.hist
Generando informe de January 2012
Generando informe resumido
Guardando información de archivo...
2039 registros en 3,00 segundos, 679/sec
root@(none):~# _
```

Una vez generado el informe deberemos de dirigirnos a un cliente y teclear en el navegador la siguiente dirección:



Index of /webalizer

Name	Last modified	Size	Description
Parent Directory		-	
ctry_usage_201201.png	10-Jan-2012 23:27	2.3K	
daily_usage_201201.png	10-Jan-2012 23:27	2.8K	
hourly_usage_201201.png	10-Jan-2012 23:27	1.7K	
index.html	10-Jan-2012 23:27	4.1K	
usage.png	10-Jan-2012 23:27	2.3K	
usage_201201.html	10-Jan-2012 23:27	58K	
webalizer.hist	10-Jan-2012 23:27	39	

Apache/2.2.16 (Ubuntu) Server at www.asir01 Port 80

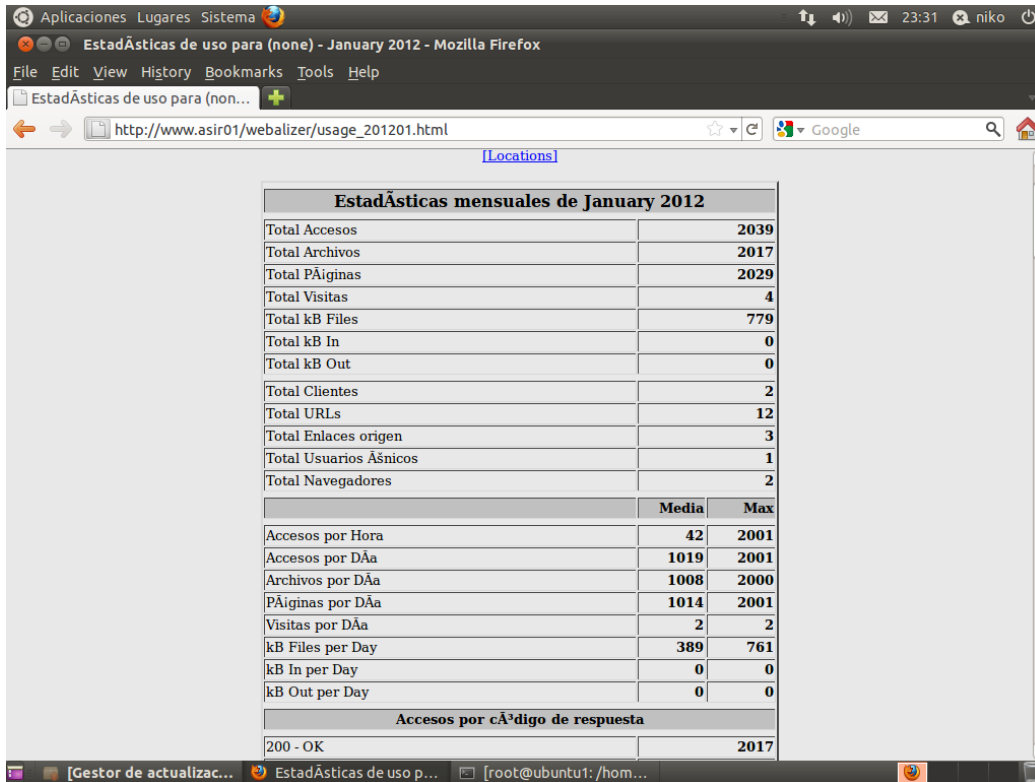
EN cada uno de los directorio podremos ver estadísticas sobre el uso del navegador; el más importante es el archivo index.html puesto que viene una gran cantidad de información.

Algunas pantallas interesantes son las siguientes:

En la siguiente pantalla podemos ver las estadísticas de uso:



En el archivo index.html podremos ver la información por meses:



A su vez también podremos ver los accesos por hora y los tipos de respuesta:

Total Navegadores		2
	Media	Max
Accesos por Hora	42	2001
Accesos por DÍA	1019	2001
Archivos por DÍA	1008	2000
Páginas por DÍA	1014	2001
Visitas por DÍA	2	2
kB Files per Day	389	761
kB In per Day	0	0
kB Out per Day	0	0
Accesos por código de respuesta		
200 - OK		2017
301 - Movido permanentemente		3
304 - No modificado		3
401 - No autorizado		1
403 - Prohibido		4
404 - No se encuentra		11

Aquí podemos ver una tabla de las estadísticas por horas del mes de junio:

Hora	Accesos		Archivos		PÁginas		kB F		kB In		kB Out	
	Media	Total	Media	Total	Media	Total	Media	Total	Media	Total	Media	Total
0	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
1	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
2	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
3	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
4	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
5	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
6	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
7	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
8	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
9	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
10	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
11	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
12	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
13	1000	2001	98,14%	1000	2000	99,16%	1000	2001	98,62%	381	761	97,74%
14	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
15	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
16	2	5	0,25%	1	3	0,15%	1	3	0,15%	1	2	0,29%
17	8	16	0,78%	5	10	0,50%	4	8	0,39%	4	8	1,01%
18	5	10	0,49%	1	2	0,10%	5	10	0,49%	2	5	0,60%
19	2	5	0,25%	1	2	0,10%	2	5	0,25%	1	2	0,24%
20	1	2	0,10%	0	0	0,00%	1	2	0,10%	0	1	0,11%
21	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%
22	0	0	0,00%	0	0	0,00%	0	0	0,00%	0	0	0,00%

Aquí podemos ver información sobre las URLs de nuestro servidor

#	Accesos	kB F	kB In	kB Out	URL		
1	2006	98,38%	763	98,00%	0	0,00%	/
2	2	0,10%	1	0,08%	0	0,00%	/privado/privado1.html
3	1	0,05%	1	0,09%	0	0,00%	/blog/
4	1	0,05%	1	0,09%	0	0,00%	/datos/
5	1	0,05%	0	0,05%	0	0,00%	/datos/datos1.html
6	1	0,05%	0	0,05%	0	0,00%	/red.html
7	1	0,05%	0	0,05%	0	0,00%	/wiki/wiki1.html
8	1	0,05%	0	0,05%	0	0,00%	/~niko/personal1.html

#	Accesos	kB F	kB In	kB Out	URL		
1	2006	98,38%	763	98,00%	0	0,00%	/
2	1	0,05%	1	0,09%	0	0,00%	/datos/
3	1	0,05%	1	0,09%	0	0,00%	/blog/
4	2	0,10%	1	0,08%	0	0,00%	/privado/privado1.html
5	1	0,05%	0	0,05%	0	0,00%	/red.html
6	1	0,05%	0	0,05%	0	0,00%	/~niko/personal1.html
7	1	0,05%	0	0,05%	0	0,00%	/wiki/wiki1.html
8	1	0,05%	0	0,05%	0	0,00%	/datos/datos1.html

#	Accesos	kB F	kB In	kB Out	URL		
1	2006	98,38%	763	98,00%	0	0,00%	/

COMANDO AB

Con este comando podemos monitorizar el estado del servidor; uno de los comandos para su ejecución es el siguiente;

```
root@(none):/etc/apache2# ab -n 1000 -c 5 -k http://localhost/
```

Ahora nos saldrá la siguiente información en la que podremos ver información como la versión del apache, el nombre del server, el tiempo de respuesta...:

```
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking localhost (be patient)

Server Software:      Apache/2.2.16
Server Hostname:     localhost
Server Port:         80

Document Path:       /
Document Length:     78 bytes

Concurrency Level:   5
Time taken for tests: 0.318 seconds
Complete requests:   1000
Failed requests:     0
Write errors:        0
Keep-Alive requests: 991
Total transferred:   389600 bytes
HTML transferred:    78000 bytes
Requests per second: 3147.95 [#/sec] (mean)
Time per request:    1.588 [ms] (mean)
Time per request:    0.318 [ms] (mean, across all concurrent requests)
:
_
```

A su vez también podemos ver información sobre las conexiones:

```
Total transferred: 389645 bytes
HTML transferred: 78000 bytes
Requests per second: 3287.00 [#/sec] (mean)
Time per request: 1.521 [ms] (mean)
Time per request: 0.304 [ms] (mean, across all concurrent requests)
Transfer rate: 1250.74 [Kbytes/sec] received

Connection Times (ms)
      min  mean[+/-sd] median  max
Connect:    0    0   0.6      0   10
Processing:  0    1   3.4      1   77
Waiting:    0    1   3.0      0   59
Total:      0    1   3.5      1   77

Percentage of the requests served within a certain time (ms)
 50%    1
 66%    1
 75%    1
 80%    1
 90%    1
 95%    2
 98%   14
 99%   15
100%   77 (longest request)
root@(none):/etc/apache2# _
```

Actividad 11: Administración servidor Web HTTPS (Apache2) en Ubuntu Server: -- Escenario: Alojamiento virtual basado en IPs.

Pasos previos:

1.-En primer lugar deberemos de añadir una nueva tarjeta de red y posteriormente configurarla con un rango de ips que corresponda al que estamos usando.

2.-En segundo lugar deberemos de configurar los CNAME del servidor DNS.

Una vez realizados los pasos anteriores realizaremos los siguientes pasos:

En primer lugar crearemos una carpeta en /var/www llamada SAD que será la raíz de nuestro servidor virtual con dicho nombre:

```
root@none):/etc/apache2/conf.d# mkdir /var/www/sad
```

A su vez en ese directorio crearemos un archivo llamado sad.html que actuara como index de dicho servidor web:

```
GNU nano 2.2.4 Archivo: /var/www/sad/sad.html
<html>
<body>
<p>Esta es la pagina de SAD de NIKO</p>
</body>
</html>
```

[5 líneas leídas]

^G Ver ayuda	^O Guardar	^R Leer Fich	^Y RePág.	^K Cortar Tex	^C Pos actual
^X Salir	^J Justificar	^W Buscar	^U Pág. Sig.	^U PegarTxt	^T Ortografía

Ahora nos dirigiremos al archivo de configuración de apache y modificaremos las líneas resaltadas en la imagen y a su vez crearemos un nuevo virtual host que nos permitirá acceder al nuevo servidor web virtual:

```

GNU nano 2.2.4          Archivo: default
<VirtualHost 10.33.1.3 *:80>
  ServerAdmin webmaster@localhost

  DocumentRoot /var/www
  <Directory />
    Options FollowSymLinks
    AllowOverride None
  </Directory>
  <Directory /var/www/>
    DirectoryIndex red.html
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
  </Directory>

  <Directory /var/www/datos>
    DirectoryIndex index.html
    Options FollowSymLinks MultiViews
    AllowOverride None
  </Directory>
[ 94 líneas leídas ]
^G Ver ayuda  ^O Guardar   ^R Leer Fich ^Y RePág.    ^K Cortar Tex ^C Pos actual
^X Salir      ^J Justificar ^W Buscar    ^V Pág. Sig. ^U PegarTxt   ^T Ortografía

```

Nuevo virtualHost:

```

GNU nano 2.2.4          Archivo: default
<VirtualHost 10.33.1.4 *:80>
DocumentRoot /var/www/sad
ServerName sad01.asir01.
ErrorLog /var/log/error.log
TransferLog /var/log/access.log
<Directory />
  Options FollowSymlinks
  AllowOverride None
</Directory>
<Directory /var/www/sad>
  DirectoryIndex sad.html
  Options Indexes FollowSymLinks MultiViews
  AllowOverride None
  Order allow,deny
  allow from all
</Directory>
</VirtualHost>
^G Ver ayuda  ^O Guardar   ^R Leer Fich ^Y RePág.    ^K Cortar Tex ^C Pos actual
^X Salir      ^J Justificar ^W Buscar    ^V Pág. Sig. ^U PegarTxt   ^T Ortografía

```

Ahora reiniciaremos el servidor apache.

Una vez reiniciado probamos a acceder con los clientes:

En primer lugar accederemos a [www.asir01.](http://www.asir01/)



ESTE ES EL SERVIDOR DE LA RED ASIR01.

Ahora accederemos a [sad01.asir01.](http://sad01.asir01/)



Esta es la pagina de SAD de NIKO

Actividad 12: Administración servidor Web HTTPS (Apache2) en Ubuntu Server --Escenario: Alojamiento virtual basado en nombres.

EN primer lugar en la carpeta /var/www crearemos 2 archivos html , uno llamado sad.html y otro asir.html, estos archivos serán la raíz para el servidor www.asir.asir01 y www.sad.asir01.

Fichero sad.html:

```
GNU nano 2.2.4 Archivo: /var/www/sad.html
<html>
<body>
<p1>ESTA ES LA PAGINA DE SAD_DE NIKO</p1>
</body>
</html>
[ 5 líneas escritas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

Fichero sri.html:

```
GNU nano 2.2.4 Archivo: /var/www/sad.html
<html>
<body>
<p1>ESTA ES LA PAGINA DE SAD_DE NIKO</p1>
</body>
</html>
[ 5 líneas escritas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora nos dirigiremos a la ruta /etc/apache2/sites-available y creamos 2 nuevos sitios uno llamado asir y otro llamado sad:

El sitio llamado sad tendrá la siguiente configuración:

```
GNU nano 2.2.4 Archivo: asir
<VirtualHost *:80 10.33.1.3>
  ServerName www.asir01.
  DocumentRoot /var/www
  DirectoryIndex asir.html
</VirtualHost>

[ 5 líneas leídas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

El sitio llamado sri tendrá la siguiente configuración:

```
GNU nano 2.2.4 Archivo: sad
<VirtualHost *:80 10.33.1.3>
  ServerName www.sad.asir01.
  DocumentRoot /var/www
  DirectoryIndex sad.html
</VirtualHost>

[ 5 líneas leídas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

Por último también deberemos de configurar la zona por defecto añadiendo la línea resaltada en la imagen:

```
GNU nano 2.2.4 Archivo: default
NameVirtualHost 10.33.1.3_
<VirtualHost *:80>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/>
        DirectoryIndex red.html
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

    <Directory /var/www/datos>
        DirectoryIndex index.html
        Options FollowSymLinks MultiViews
        [ 70 líneas escritas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora en el apache2.conf deberemos de añadir las siguientes líneas:

```
GNU nano 2.2.4 Archivo: apache2.conf
LogFormat "%{User-agent}i" agent

# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.

# Include generic snippets of statements
Include conf.d/

# Include the virtual host configurations:
include sites-enabled/
include sites-enabled/sad
include sites-enabled/asir
-
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

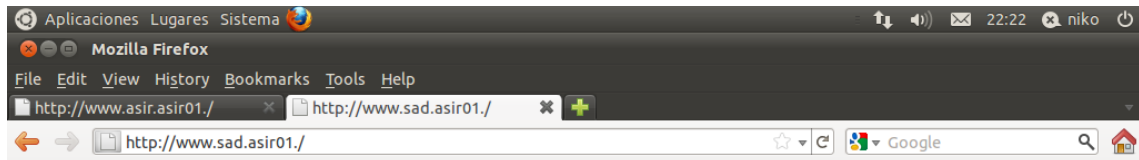
Ahora activaremos las zonas con los comandos de la imagen:

```
root@(none):/etc/apache2/sites-available# a2ensite asir
Enabling site asir.
Run '/etc/init.d/apache2 reload' to activate new configuration!
root@(none):/etc/apache2/sites-available# a2ensite sad
Enabling site sad.
Run '/etc/init.d/apache2 reload' to activate new configuration!
root@(none):/etc/apache2/sites-available#
```

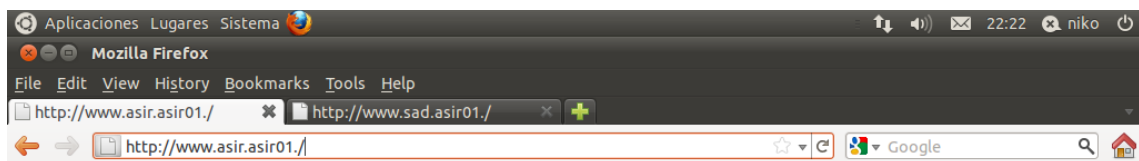
Por ultimo reiniciamos nuestro servidor apache

Una vez reiniciado probamos el acceso con un cliente:

Prueba de acceso a [www.sad.asir01.](http://www.sad.asir01/)



Prueba de acceso a [www.sri.asir01.](http://www.sri.asir01/)



Actividad 13: Administración servidor Web HTTPS (Apache2) en Ubuntu Server --Escenario: Alojamiento virtual basado en puertos.

En primer lugar modificaremos el archivo `ports.conf` y añadimos las líneas remarcadas en la imagen para que nuestro servidor resuelva las peticiones al puerto 8080:

```
GNU nano 2.2.4 Archivo: ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default
# This is also true if you have upgraded from before 2.2.9-3 (i.e. from
# Debian etch). See /usr/share/doc/apache2.2-common/NEWS.Debian.gz and
# README.Debian.gz

NameVirtualHost 10.33.1.3:80
Listen 80
NameVirtualHost 10.33.1.3:8080
Listen 8080

<IfModule mod_ssl.c>
# If you add NameVirtualHost *:443 here, you will also have to change
# the VirtualHost statement in /etc/apache2/sites-available/default-ssl
# to <VirtualHost *:443>
# Server Name Indication for SSL named virtual hosts is currently not
# supported by MSIE on Windows XP.
Listen 443
</IfModule>

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora crearemos una zona llamada `asir`, la cual tendrá su raíz en la ruta `/var/www` y cuyo archivo por defecto será el archivo `asir.html`. A su vez también le indicamos que el puerto de esta zona será el 80. Podremos acceder a esta zona introduciendo en el navegador `www.asir.asir01`:

```
GNU nano 2.2.4 Archivo: asir
<VirtualHost 10.33.1.3:80>
    ServerName www.asir.asir01
    DocumentRoot /var/www
    DirectoryIndex asir.html
</VirtualHost>

[ 5 líneas leídas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora crearemos una zona llamada sad, la cual tendrá su raíz en la ruta /var/www y cuyo archivo por defecto será el archivo asir.html. A su vez también le indicamos que el puerto de esta zona será el 8080. Podremos acceder a esta zona introduciendo en el navegador www.sad.asir01.:

```
GNU nano 2.2.4 Archivo: sad
<VirtualHost 10.33.1.3:8080>
    ServerName www.sad.asir01
    DocumentRoot /var/www
    DirectoryIndex sad.html
</VirtualHost>

[ 5 líneas leídas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

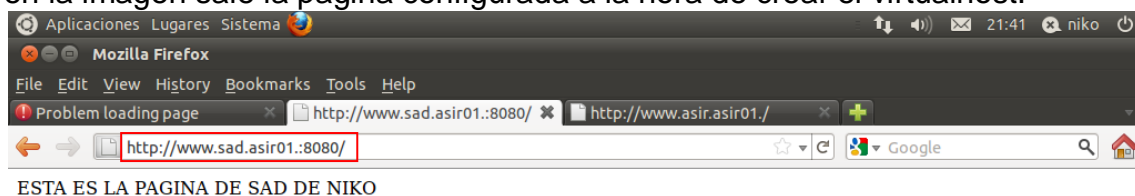
Ahora activamos las zonas creadas:

```
root@(none):/var/log/apache2# a2ensite asir
root@(none):/var/log/apache2# a2ensite sad
```

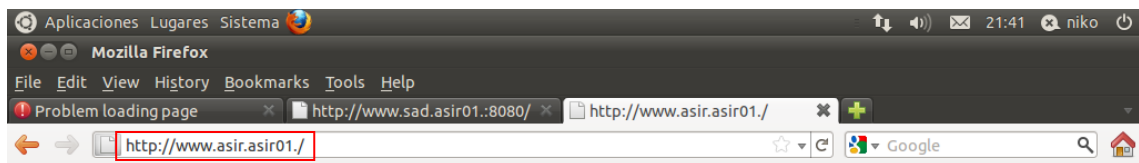
Por último reiniciamos nuestro servidor:

```
root@(none):/etc/apache2/sites-available# service apache2 restart
```

Ahora comprobaremos el funcionamiento de los configurado desde un cliente, en primer lugar accederemos a la dirección www.sad.asir01:8080, como se ve en la imagen sale la pagina configurada a la hora de crear el virtualhost:



Por último comprobaremos el funcionamiento de los configurado desde un cliente, en primer lugar accederemos a la dirección [www.asir.asir01.](http://www.asir.asir01/) , como se ve en la imagen sale la página configurada a la hora de crear el virtualhost:



ESTA ES LA PAGINA DE ASIR DE NIKO

Actividad 14: Instalación y configuración de un servidor web HTTP Apache2 mediante Webmin.

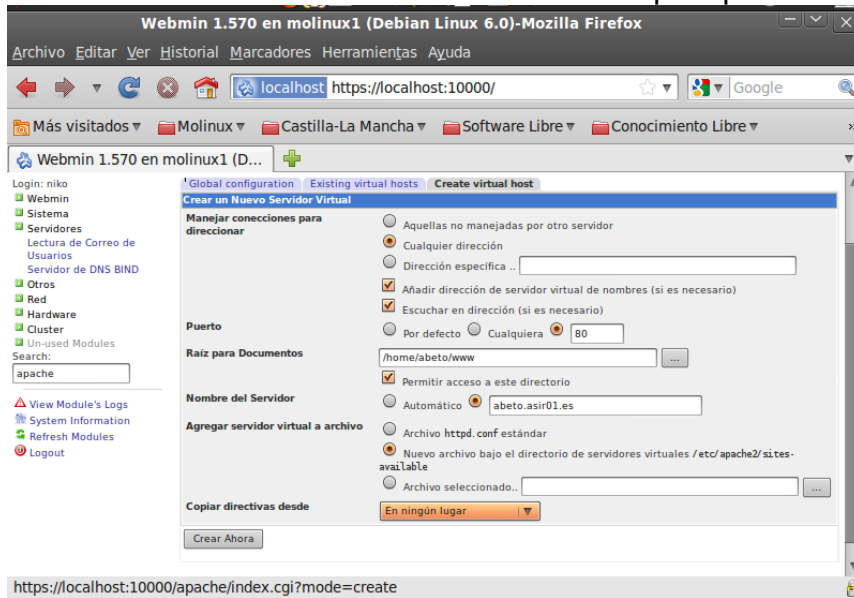
En primer lugar crearemos los usuarios abeto :

```
root@molinux1: /home/niko
Archivo Editar Ver Buscar Terminal Ayuda
niko@molinux1:~$ sudo su
[sudo] password for niko:
root@molinux1:/home/niko# adduser abeto
Añadiendo el usuario `abeto' ...
Añadiendo el nuevo grupo `abeto' (1001) ...
Añadiendo el nuevo usuario `abeto' (1001) con grupo `abeto' ...
Creando el directorio personal `/home/abeto' ...
Copiando los ficheros desde `/etc/skel' ...
Introduzca la nueva contraseña de UNIX:
Vuelva a escribir la nueva contraseña de UNIX:
No se ha proporcionado ninguna contraseña
Introduzca la nueva contraseña de UNIX:
Vuelva a escribir la nueva contraseña de UNIX:
passwd: contraseña actualizada correctamente
Changing the user information for abeto
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
¿Es correcta la información? [S/n] s
root@molinux1:/home/niko#
```

Y chopo:

```
root@molinux1:/home/niko# adduser chopo
Añadiendo el usuario `chopo' ...
Añadiendo el nuevo grupo `chopo' (1002) ...
Añadiendo el nuevo usuario `chopo' (1002) con grupo `chopo' ...
Creando el directorio personal `/home/chopo' ...
Copiando los ficheros desde `/etc/skel' ...
Introduzca la nueva contraseña de UNIX:
Vuelva a escribir la nueva contraseña de UNIX:
passwd: contraseña actualizada correctamente
Changing the user information for chopo
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
¿Es correcta la información? [S/n]
root@molinux1:/home/niko#
```


Ahora crearemos un nuevo virtual host, para ello nos dirigimos a la pestaña create virtual host e introducimos los valores que aparecen en la imagen:



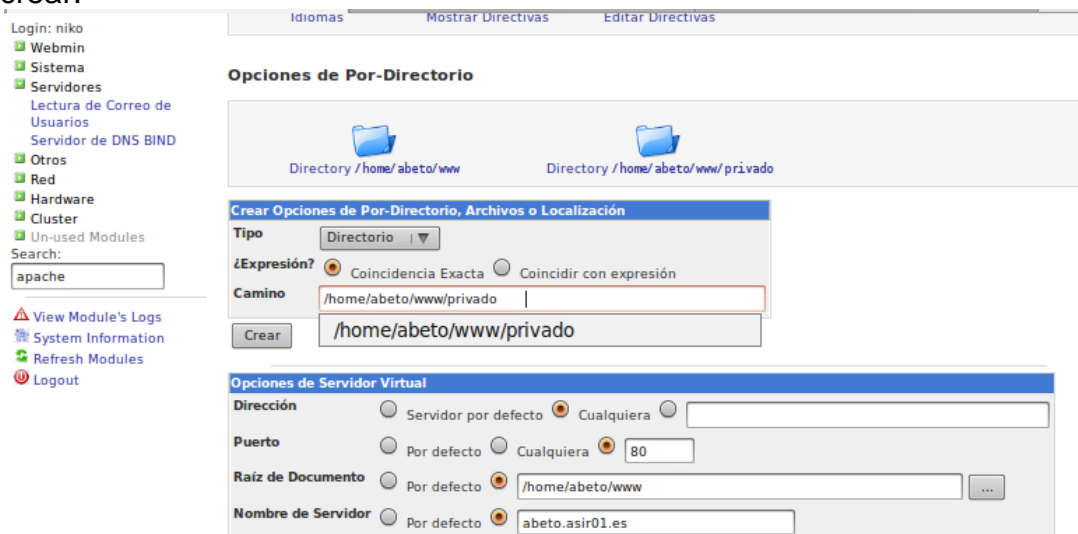
Ahora crearemos un fichero llamado indice.html en la raíz de nuestro servidor:



Ahora haremos que el index por defecto sea el archivo índice.html creado previamente:

```
DocumentRoot "/home/abeto/www"
ServerName abeto.asir01.es
<Directory "/home/abeto/www">
DirectoryIndex indice.html
Options FollowSymlinks MultiViews
allow from all
</Directory>
```

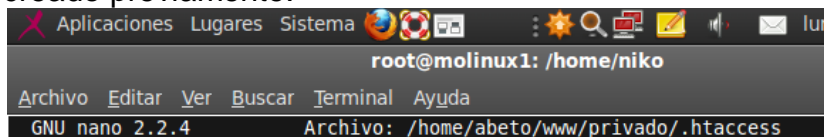
Ahora crearemos el directorio privado. En la siguiente imagen pulsamos sobre crear:



Ahora crearemos el archivo que contendrá el nombre de usuario y la contraseña necesarias para restringir el acceso al usuario abeto:

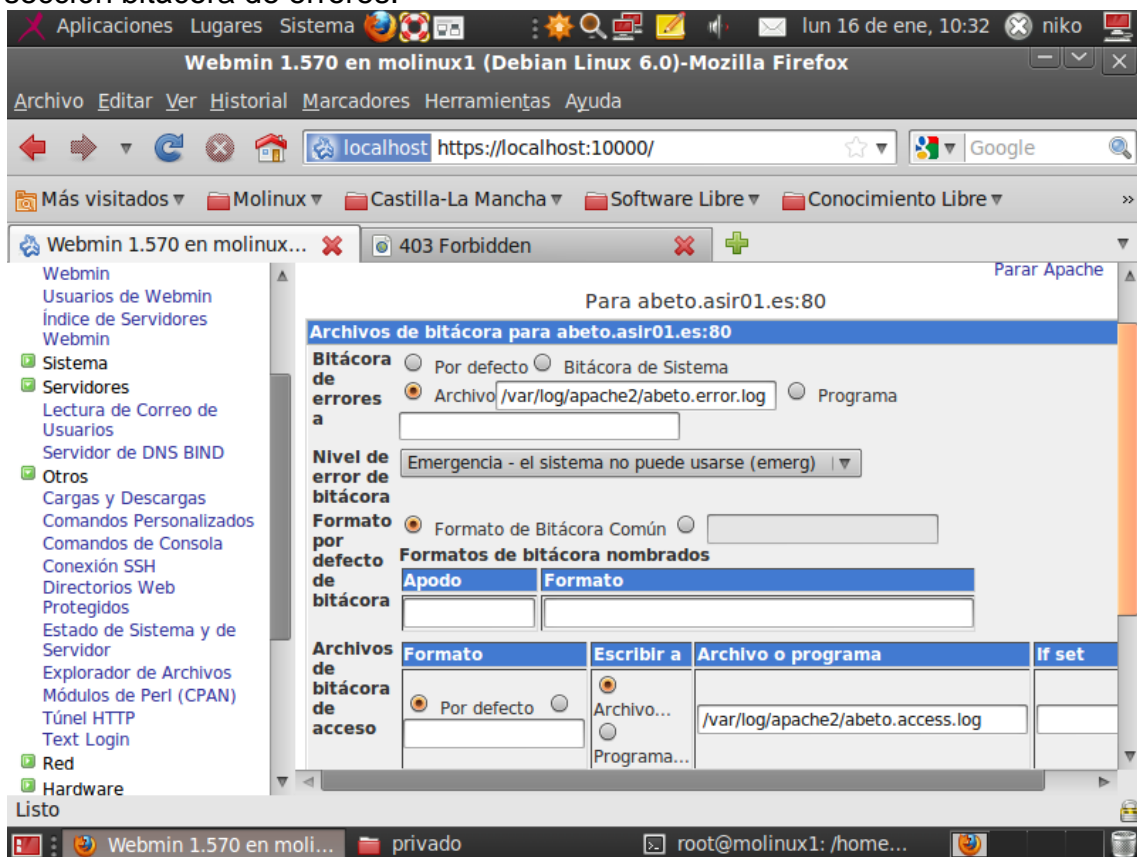
```
root@molinux1:/home/niko# htpasswd -c /home/abeto/.htpasswd abeto
New password:
Re-type new password:
Adding password for user abeto
```

Ahora deberemos de crear el fichero .htaccess con la configuración de la imagen para poder restringir el acceso a todos lo usuario excepto a abeto y a todos los equipos excepto a local y a la dirección ip 10.33.1.5. en este fichero también deberemos de indicar la ruta de nuestro fichero .htpasswd que hemos creado previamente:

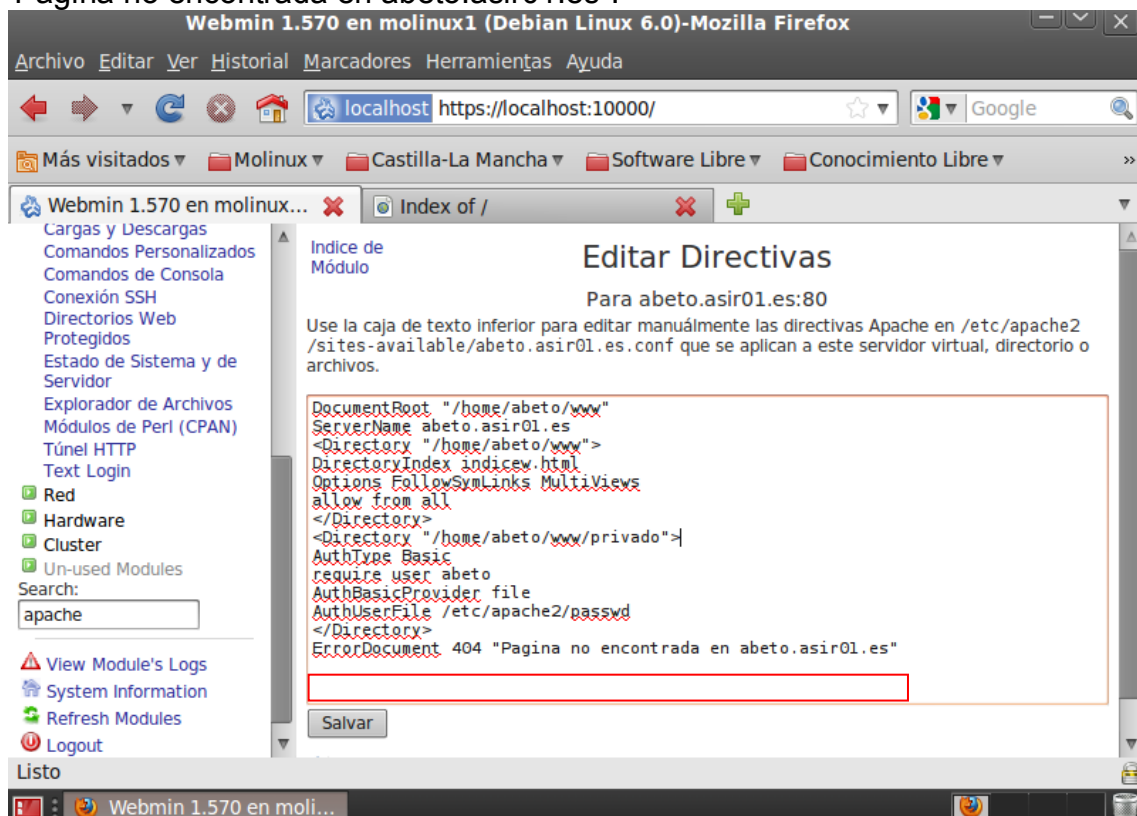


```
DirectoryIndex indice.html
Options Indexes FollowSymlinks MultiViews
Order allow,deny
allow from localhost 127.0.0.1
allow from 10.33.1.5
AuthName "Acceso privado"
AuthType Basic
AuthUserFile /home/abeto/.htpasswd
Require user abeto
```

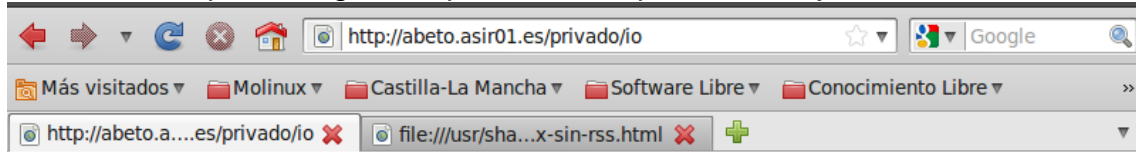
Ahora cambiaremos la ruta de los log de apache 2, para ello nos dirigimos a la sección bitácora de errores:



Ahora en la sección editar directivas añadimos el error 404 y la respuesta "Página no encontrada en abeto.asir01.es":

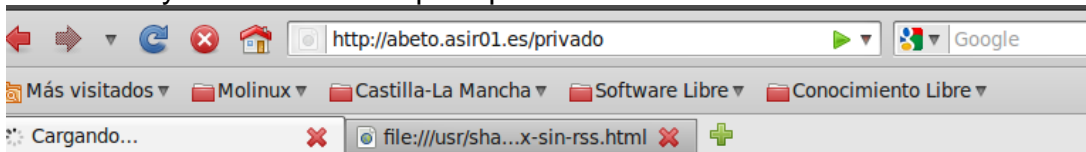


Ahora comprobaremos el funcionamiento de lo realizado en los pasos anteriores, en primer lugar comprobaremos que el mensaje de error funciona:

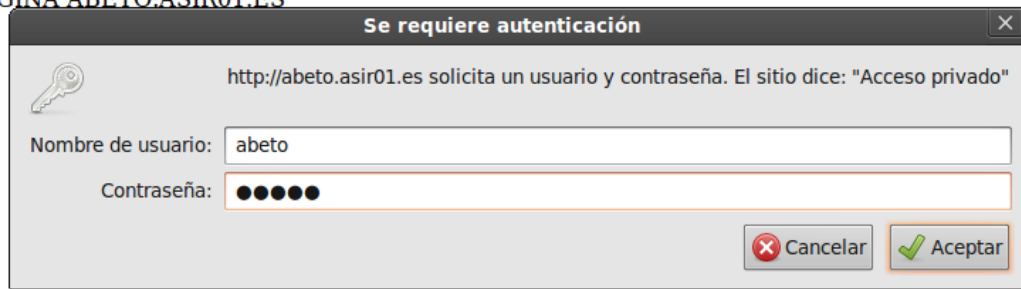


Pagina no encontrada en abeto.asir01.es

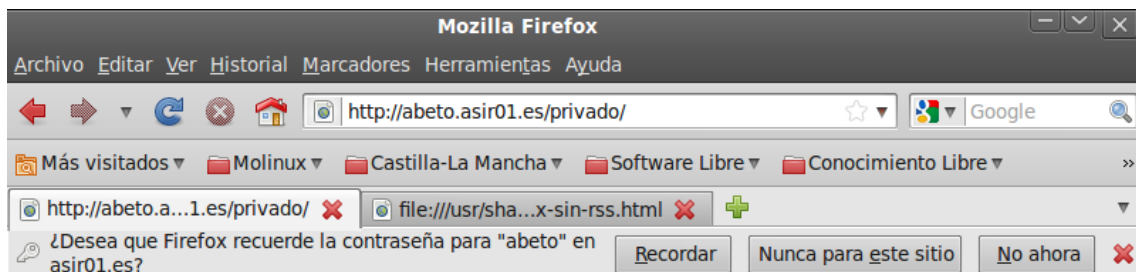
Ahora comprobaremos que el directorio configurado con .htaccess nos requiere un usuario y una contraseña para poder acceder:



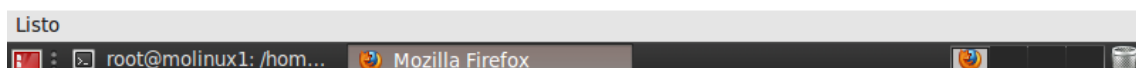
PAGINA ABETO:ASIR01.ES



Y que una vez introducido el usuario y contraseña podremos ver la pagina web:

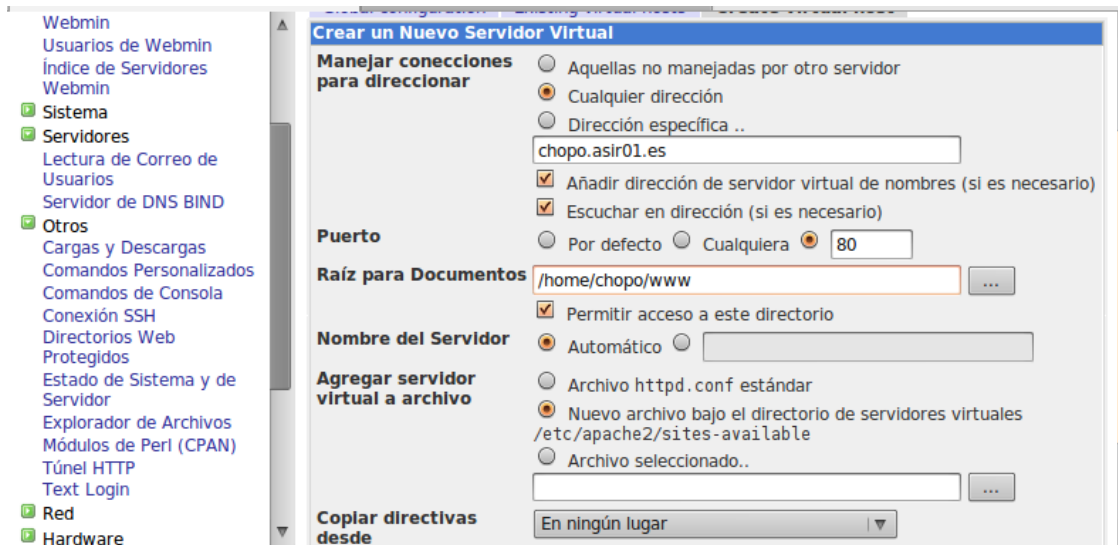


PAGINA PRIVADA DE ABETO:ASIR01.ES

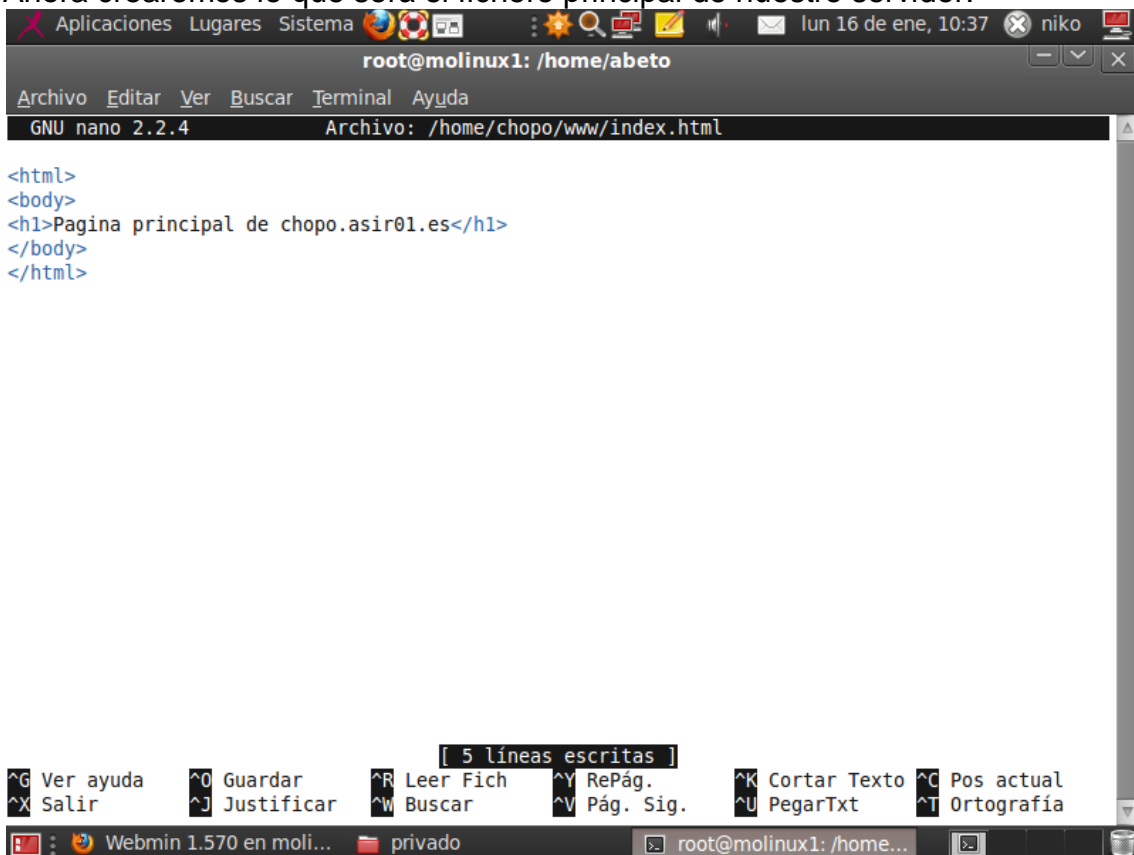


Configuración de chopo.asir01.es:

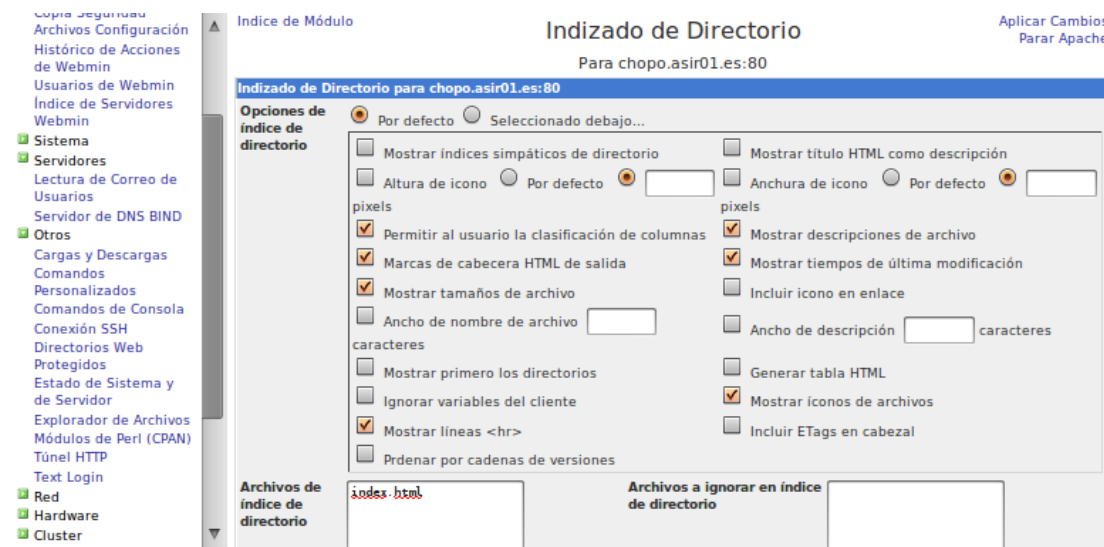
En primer lugar crearemos el virtualhost llamado chopo.asir01.es:



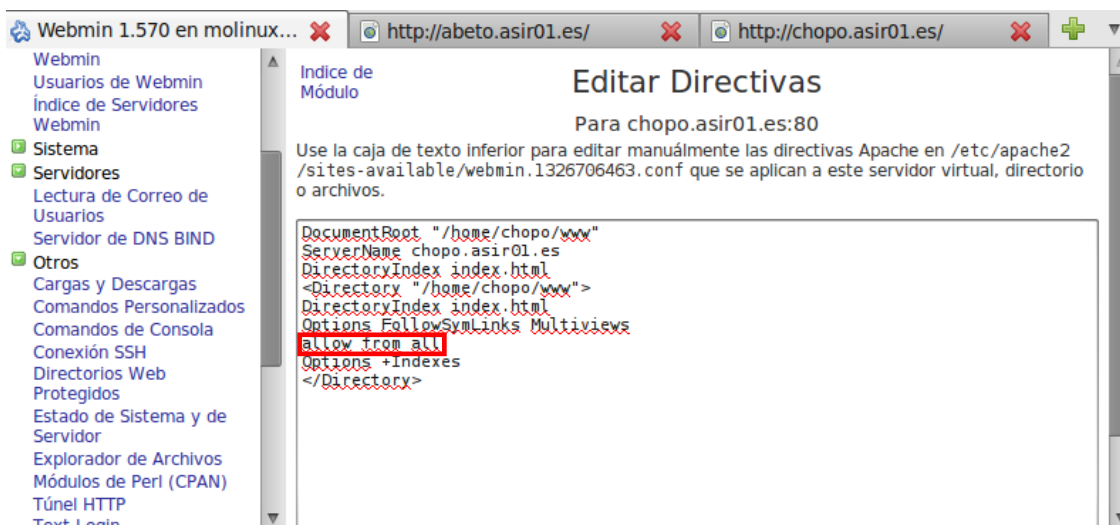
Ahora crearemos lo que será el fichero principal de nuestro servidor:



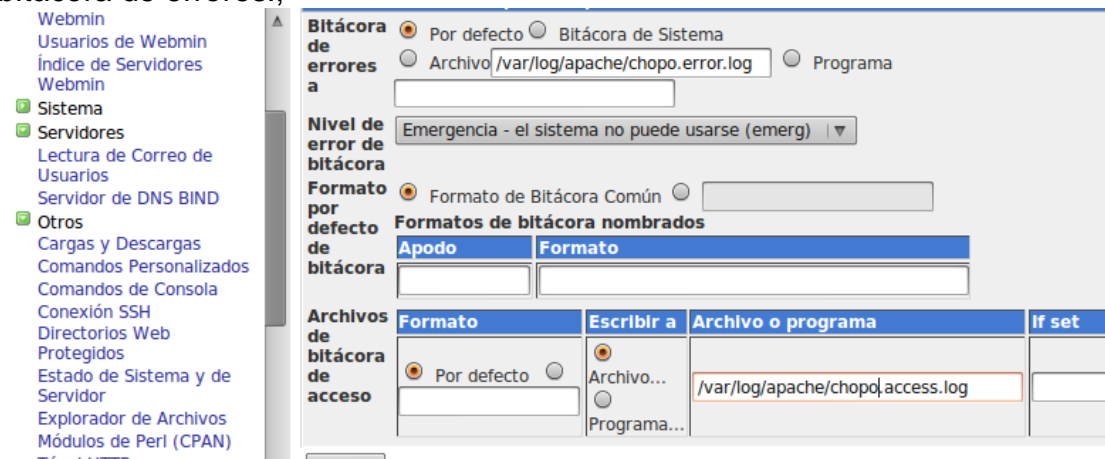
Ahora nos dirigimos a la sección Indizado de directorio y allí configuramos las opciones que aparecen en pantalla para que se muestre por defecto el archivo index.html y no se muestre el listado del directorio raíz:



Para permitir el acceso a todos los usuarios en la opción editar directivas introducimos la opción allow from all:



Ahora modificaremos la ruta de los log de errores, para ello nos dirigimos a la bitácora de errores;



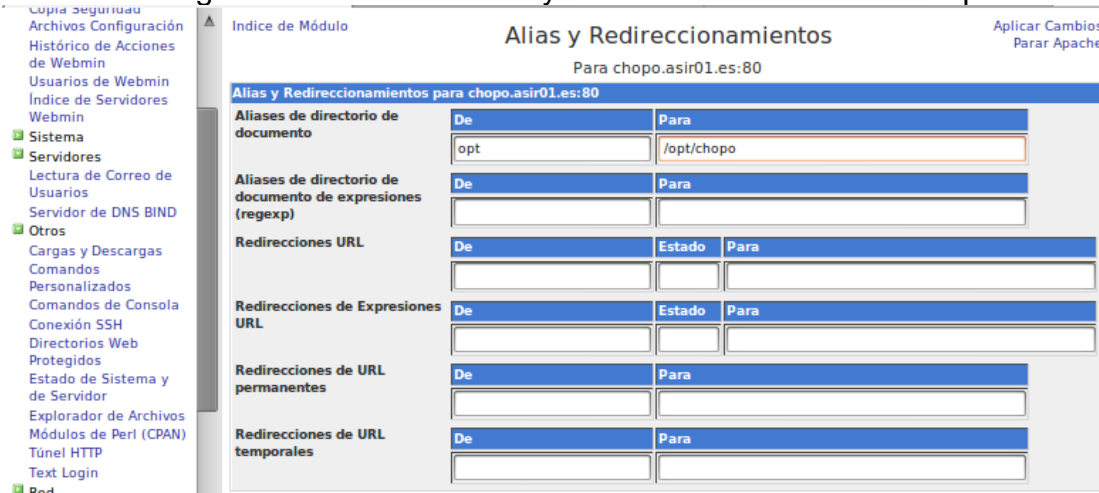
Ahora en la opción manejo de errores introducimos el error 404:



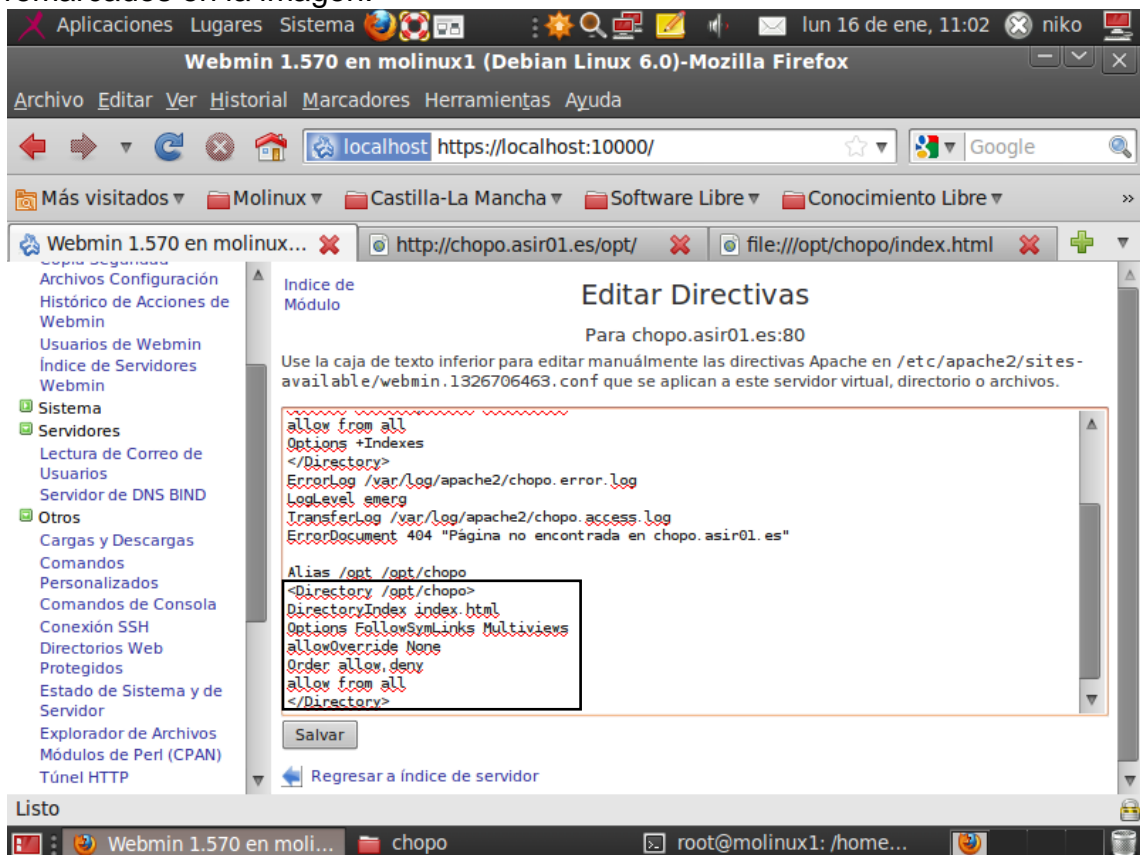
Ahora crearemos el directorio /opt/chopo

```
root@molinux1:/home/abeto# mkdir /opt/chopo
```

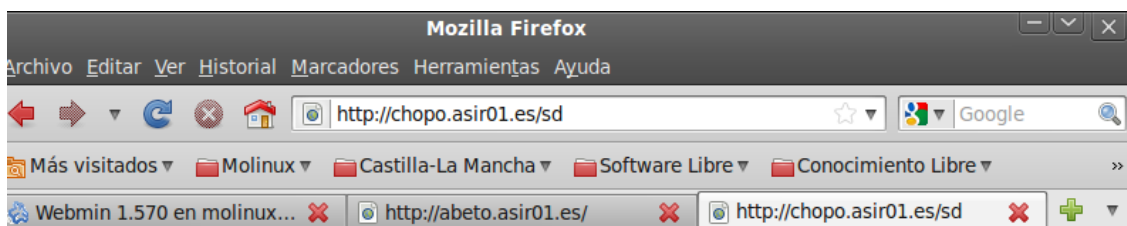
Ahora nos dirigimos a la sección alias y introducimos los valores de pantalla:



Por último nos dirigimos a editar directivas e introducimos los valores remarcados en la imagen:

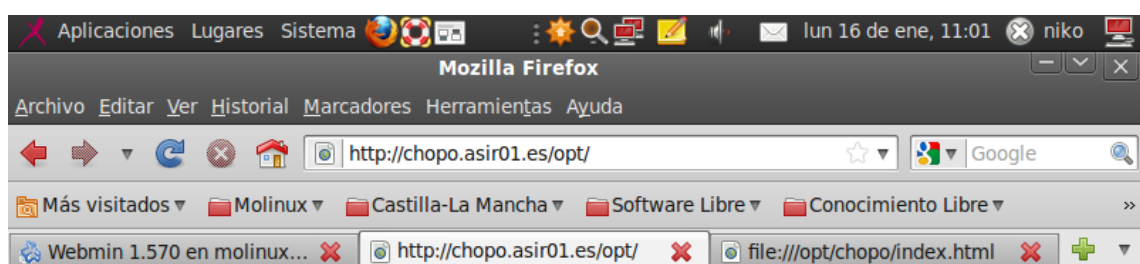


Ahora realizaremos una prueba del funcionamiento de la introducción de una página errónea:



Página no encontrada en chopo.asir01.es

Ahora probaremos ha acceder al directorio enlazado; es decir el opt:



PAGINA OPT DE chopo.asir01.es

Actividad 15: Administración servidor web HTTPS (Apache2) en Ubuntu Server: -- mod_ssl, default_ssl - certificados digitales.

En primer lugar procederemos a activar el modulo ssl de nuestro apache, para ello introducimos el siguiente comando:

```
root@(none):/var/www/datos# aZenmod ssl
Enabling module ssl.
See /usr/share/doc/apache2.2-common/README.Debian.gz on how to configure SSL and
create self-signed certificates.
Run '/etc/init.d/apache2 restart' to activate new configuration!
```

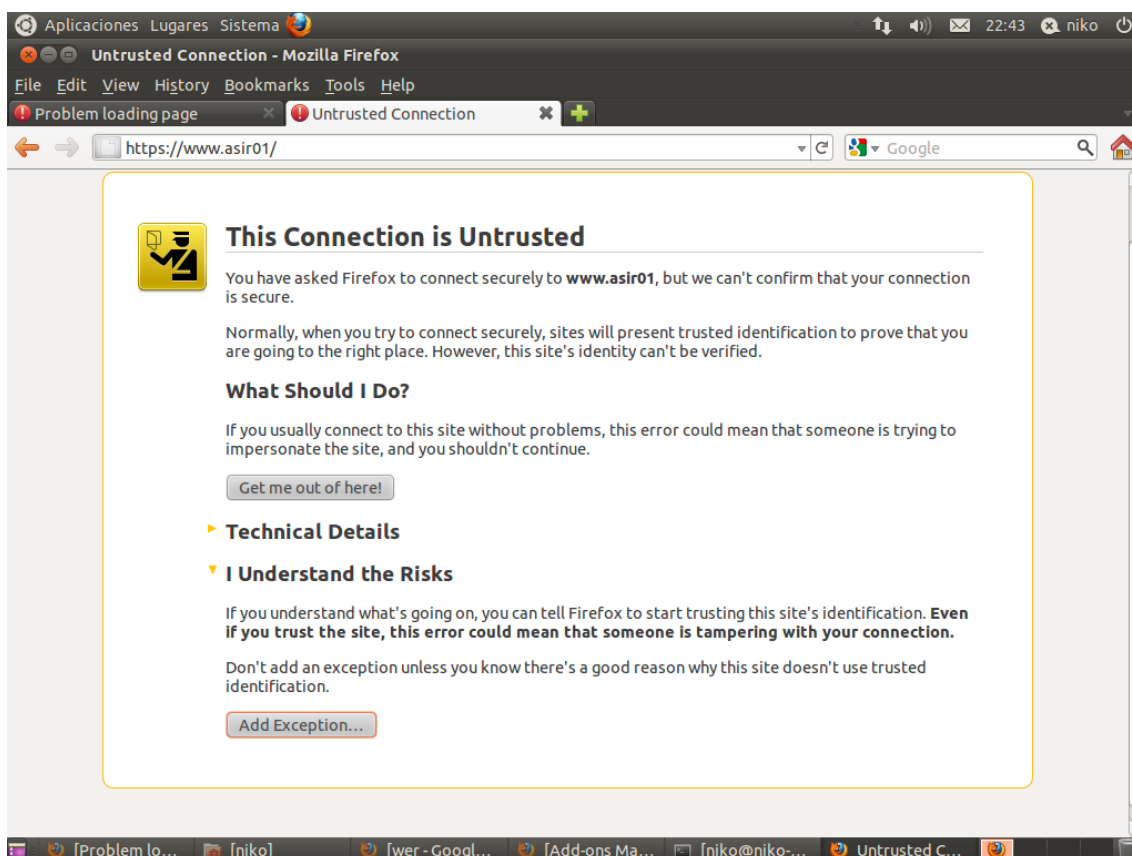
Ahora también activaremos la zona default-ssl que es la zona por defecto SSL de nuestro servidor:

```
root@(none):/etc/apache2/sites-available# aZensite default-ssl
Enabling site default-ssl.
Run '/etc/init.d/apache2 reload' to activate new configuration!
```

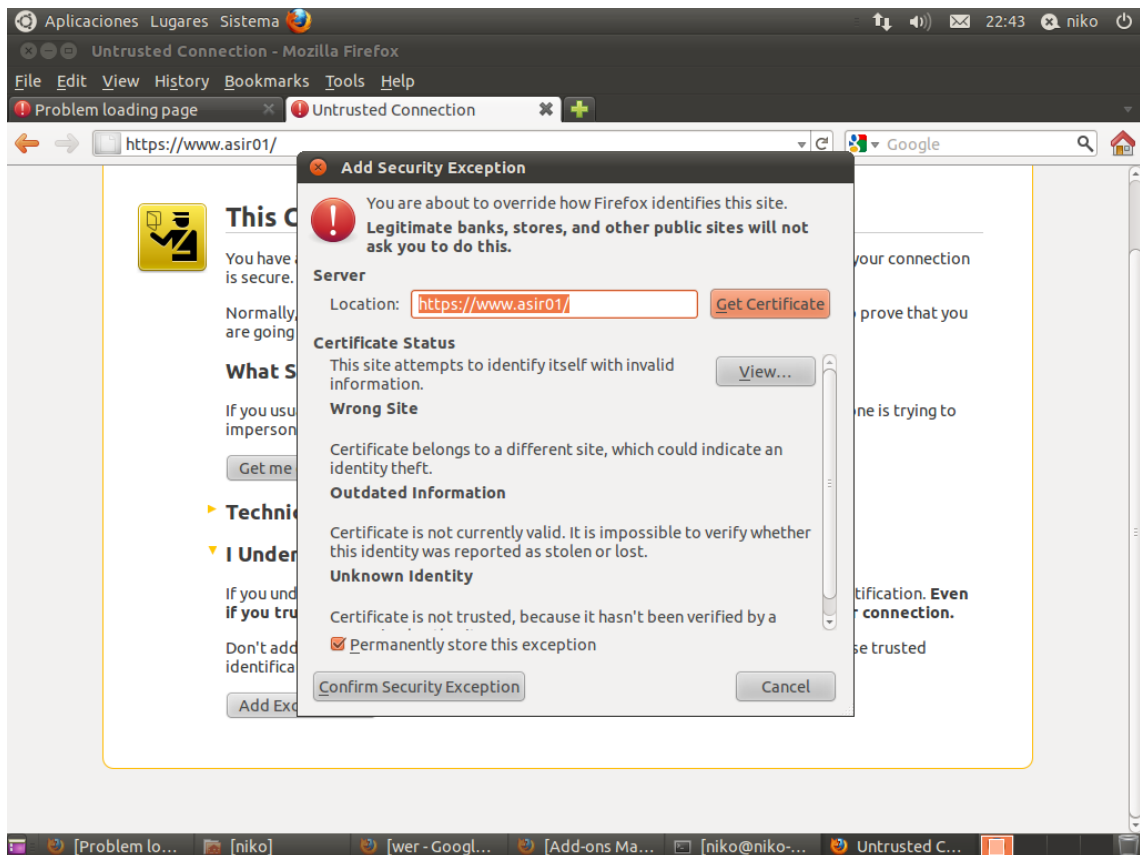
Ahora reiniciaremos el servicio de apache2:

```
root@(none):~# service apache2 restart
```

Ahora desde un cliente introduciremos en el navegador la dirección `https://www.asir01.` , como se aprecia en la imagen, la pagina posee un certificado, para acceder a la imagen deberemos de pulsar el botón añadir excepción:



Ahora confirmaremos la excepción del seguridad, una vez realizado esto podremos ver nuestra página web:



Ahora desactivaremos el sitio default-ssl:

```
root@(none):/etc/apache2/sites-available# a2dissite default-ssl
Site default-ssl disabled.
Run '/etc/init.d/apache2 reload' to activate new configuration!
root@(none):/etc/apache2/sites-available#
```

Ahora deberemos de crear un nuevo sitio que también será controlado mediante certificados, a su vez configuraremos los log de errores que se almacenara en la ruta especificada en la imagen:

```
GNU nano 2.2.4 Archivo: seguro
<VirtualHost *:443>
  ServerName www.seguro.asir01.
  DocumentRoot /var/www
  DirectoryIndex red.html
</VirtualHost>
ErrorLog /var/log/apache2/seguro.error.log
CustomLog /var/log/apache2/seguro.access_log combined
```

Una vez creado el sitio crearemos los certificados, para ello deberemos de introducir los siguientes comandos:

1.- EN primer lugar creamos la key:

```
OpenSSL> genrsa -des3 -out cer.key 4096
Generating RSA private key, 4096 bit long modulus
.....++
.....++
e is 65537 (0x10001)
Enter pass phrase for cer.key:
Verifying - Enter pass phrase for cer.key:
```

2.- Una vez creada la key crearemos el archivo cer.csr, para ello introducimos el comando de la imagen y rellenamos el cuestionario que nos aparece

```
OpenSSL> req -new -key cer.key -out cer.csr
Enter pass phrase for cer.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:ES
State or Province Name (full name) [Some-State]:Ciudad Real
Locality Name (eg, city) []:Lasolana
Organization Name (eg, company) [Internet Widgits Pty Ltd]:ASIR01
Organizational Unit Name (eg, section) []:asir
Common Name (eg, YOUR name) []:127.0.0.1
Email Address []:niko@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
OpenSSL>
```

3.- Ahora por ultimo firmamos el certificado, para ello introducimos el siguiente comando:

```
OpenSSL> x509 -req -days 365 -in cer.csr -signkey cer.key -out cer.crt
Signature ok
subject=/C=ES/ST=Ciudad Real/L=Lasolana/O=ASIRO1/OU=asir/CN=127.0.0.1/emailAddress=niko@gmail.com
Getting Private key
Enter pass phrase for cer.key:
OpenSSL>
```

Ahora insertaremos las siguientes líneas en la zona creada para poder crear un sitio https:

```
GNU nano 2.2.4 Archivo: seguro
<VirtualHost *:443>
    ServerName www.seguro.asir01.
    DocumentRoot /var/www
    DirectoryIndex red.html
ErrorLog /var/log/apache2/seguro.error.log
CustomLog /var/log/apache2/seguro.access.log combined
SSLEngine On
SSLCertificateFile /etc/apache2/sites-available/cert.crt
SSLCertificateKeyFile /etc/apache2/sites-available/cer.key
</VirtualHost>

[ 10 líneas leídas ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Repág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

Ahora reiniciamos y accedemos desde el cliente al equipo servidor con una dirección https, en nuestro caso será www.seguro.asir01.

```
root@(none):/etc/apache2/sites-available# service apache2 restart
* Restarting web server apache2
... waiting .Apache/2.2.16 mod_ssl/2.2.16 (Pass Phrase Dialog)
Some of your private key files are encrypted for security reasons.
In order to read them you have to provide the pass phrases.

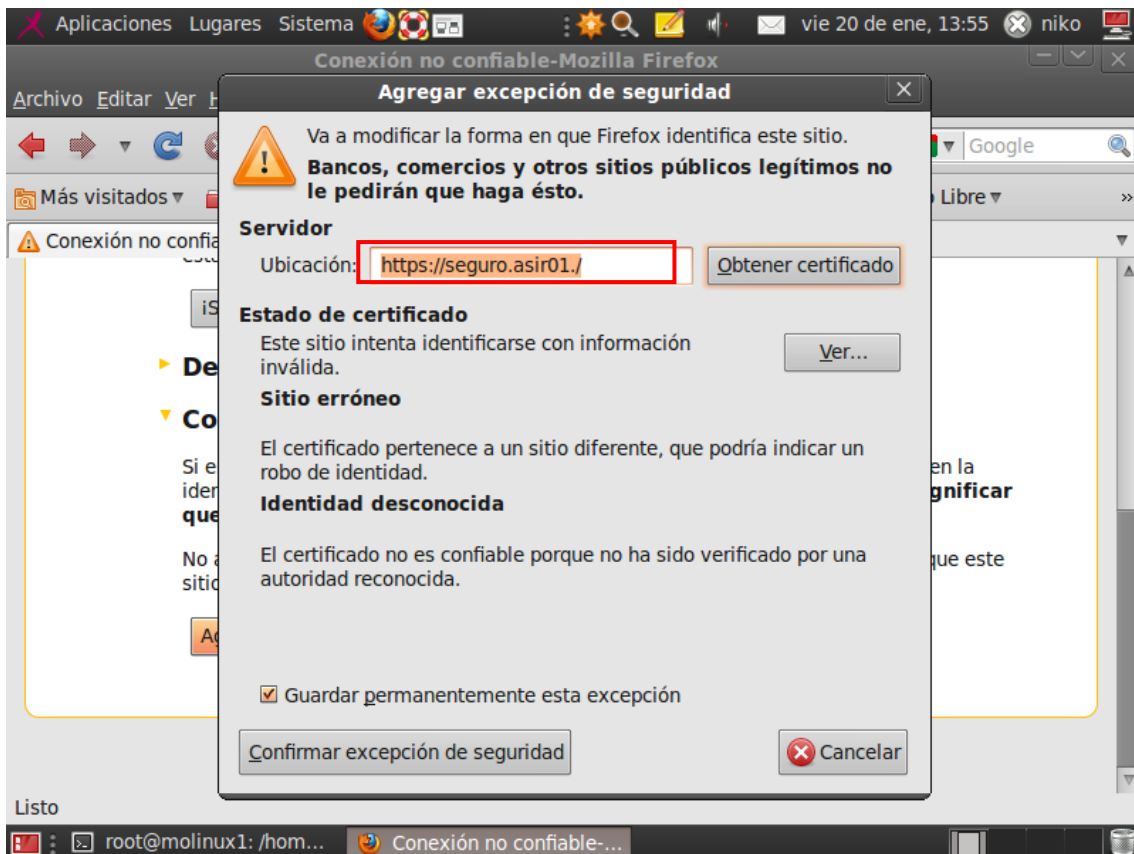
Server www.seguro.asir01.:443 (RSA)
Enter pass phrase:

OK: Pass Phrase Dialog successful.
root@(none):/etc/apache2/sites-available#
```

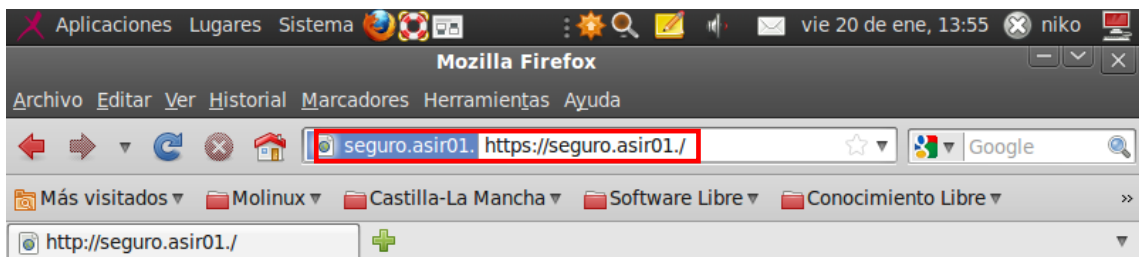
Ahora accedemos a la dirección indicada:



Ahora aceptamos el certificado:



Ahora podremos ver el contenido de nuestra página:



**ESTE ES EL SITIO SEGURO DE LA RED
ASIR01.**

